



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR – TE141599

**PENALAAAN OPTIMAL POWER SYSTEM STABILIZER
(PSS) MENGGUNAKAN ALGORITMA RANDOM DRIFT
PARTICLE SWARM OPTIMIZATION (RDPSO)
DENGAN PERMODELAN SMIB PADA PEMBANGKIT
LISTRIK**

Okto Fenno
NRP 2212 100 135

Dosen Pembimbing
Prof. Dr. Ir. Adi Soeprijanto, MT.
Ir. Ni Ketut Aryani, MT.

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2016



ITS
Institut
Teknologi
Sepuluh Nopember

FINAL PROJECT – TE141599

**OPTIMIZING POWER SYSTEM STABILIZER
PARAMETERS USING RANDOM DRIFT PARTICLE
SWARM OPTIMIZATION (RDPSO) ALGORITHM FOR
POWER PLANT IN SMIB MODELLING**

Okto Fenno
NRP 2212 100 135

Supervisor
Prof. Dr. Ir. Adi Soeprijanto, MT.
Ir. Ni Ketut Aryani, MT.

ELECTRICAL ENGINEERING DEPARTEMENT
Faculty of Industrial Technology
Sepuluh Nopember Institute of Technology
Surabaya 2016

**PENALAAN OPTIMAL POWER SYSTEM STABILIZER
(PSS) MENGGUNAKAN ALGORITMA RANDOM DRIFT
PARTICLE SWARM OPTIMIZATION (RDPSO) DENGAN
PERMODELAN SMIB PADA PEMBANGKIT LISTRIK**

TUGAS AKHIR

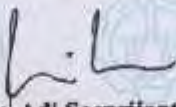
**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik**

Pada

**Bidang Studi Teknik Sistem Tenaga
Jurusan Teknik Elektro
Institut Teknologi Sepuluh Nopember**

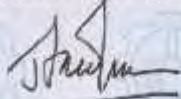
Menyetujui :

Dosen Pembimbing I



Prof. Dr. Ir. Adi Soeprijanto, MT.
NIP. 1964 04 05 1990 02 1001

Dosen Pembimbing II



Ir. Ni Ketut Arvani, MT.
NIP. 1965 09 01 1991 03 2002



PENALAAN OPTIMAL *POWER SYSTEM STABILIZER* (PSS) MENGGUNAKAN ALGORITMA *RANDOM DRIFT PARTICLE SWARM OPTIMIZATION* (RDPSO) DENGAN PERMODELAN SMIB PADA PEMBANGKIT LISTRIK

ABSTRAK

Nama Mahasiswa : Okto Fenno
NRP : 2212 100 135
Dosen Pembimbing 1 : Prof. Dr. Ir. Adi Soeprijanto, MT.
NIP : 1964 04 05 1990 02 1001
Dosen Pembimbing 2 : Ir. Ni Ketut Aryani, MT.
NIP : 1965 09 01 1991 03 2002

Pembangkit Listrik Tenaga Uap (PLTU) Pacitan merupakan salah satu dari realisasi program percepatan pembangunan pembangkit listrik 10 MW di Indonesia. Pasca gangguan di PLTU Pacitan pada tahun 2014, Tim Peneliti Pusat Pengaturan Beban Jawa-Bali memperkirakan bahwa terjadinya perubahan rasio permintaan beban dan total daya yang dihasilkan pada sistem ini menyebabkan terjadinya osilasi pada sistem. Osilasi yang terjadi pada pembangkit Pacitan diduga juga disebabkan oleh nilai redaman yang jelek dan pembangkit menyuplai beban yang jaraknya cukup jauh dari lokasi pembangkit. Salah satu upaya untuk mengatasi osilasi adalah dengan menambahkan *Power System Stabilizer* pada pembangkit. Pada tugas akhir ini, dilakukan simulasi penalaan parameter PSS yang diselesaikan dengan menggunakan *Random Drift Particle Swarm Optimization* (RDPSO), *Zeq* sistem didapat dengan melakukan reduksi jaringan menggunakan metode *losses concept*. Dari hasil simulasi, didapatkan bahwa *overshoot* dan *settling time* dari respon frekuensi, dan respon sudut rotor pada model SMIB dengan PSS yang tertala dengan RDPSO memiliki nilai yang paling rendah dibandingkan dengan model SMIB tanpa menggunakan PSS, dan dengan PSS yang belum dioptimalkan. Frekuensi osilasi mode *local plant* berhasil dihilangkan. Frekuensi osilasi yang tersisa pada simulasi yang dilakukan adalah frekuensi osilasi *interarea*.

Kata Kunci : RDPSO, PSS, SMIB, *Losses Concept*

OPTIMIZING POWER SYSTEM STABILIZER PARAMETERS USING RANDOM DRIFT PARTICLE SWARM OPTIMIZATION ALGORITHM FOR POWER PLANT IN SMIB MODELLING

ABSTRACT

Name of Student : Okto Fenno
Registration Number : 2212100135
Consultive Lecturer 1 : Prof. Dr. Ir. Adi Soeprijanto, MT.
ID : 1964 04 05 1990 02 1001
Consultive Lecturer 2 : Ir. Ni Ketut Aryani, MT.
ID : 1965 09 01 1991 03 2002

Pacitan Steam Power Plant is one of the realization of the accelerated development of 10 MW power plant in Indonesia. Post-interference in Pacitan power plant in 2014, the Research Team of Load Control Center Java - Bali estimated that the change of load demand ratio and the total power generated in this system causes oscillation in the system. Oscillation occurs in Pacitan plant suspected to be caused by poor attenuation value and power supply load a long distance from the plant site . One effort to overcome the oscillation is by adding Power System Stabilizer on plants. In this final project, The tuning of PSS parameters is simulated on Pacitan Power Plan. This case is solved using Random Drift Particle Swarm Optimization (RDPSO) algorithm, *Zeq* on system is achieved by doing network reduction using losses concept. Taken from the simulation result, it is known that SMIB model attached with optimized PSS on both cases has lower overshoot of frequency and rotor angle response than the other model (SMIB with unoptimized PSS and SMIB without PSS). Based on simulation result, the local area oscillation mode is successfully damped. The remaining oscillation within system model are only interarea oscillation mode.

Keywords : RDPSO, PSS, SMIB, Losses Concept

DAFTAR ISI

| | |
|--|--------------|
| JUDUL | i |
| PERNYATAAN KEASLIAN TUGAS AKHIR | v |
| LEMBAR PENGESAHAN | vii |
| ABSTRAK | ix |
| ABSTRACT | xi |
| KATA PENGANTAR | xiii |
| DAFTAR ISI | xv |
| DAFTAR GAMBAR | xix |
| DAFTAR TABEL | xxi |
| DAFTAR NOMENKLATUR | xxiii |
| | |
| BAB I PENDAHULUAN | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Tujuan..... | 2 |
| 1.3 Perumusan Masalah | 2 |
| 1.4 Batasan Masalah | 3 |
| 1.5 Metode..... | 3 |
| 1.5.1 Tahapan Permodelan Sistem | 4 |
| 1.5.2 Pengujian Sistem | 4 |
| 1.5.3 Tahap Optimasi Parameter Kontrol PSS..... | 4 |
| 1.5.4 Tahap analisis | 5 |
| 1.6 Sistematika Penulisan..... | 5 |
| 1.7 Relevansi dan Manfaat | 6 |
| | |
| BAB II DASAR TEORI..... | 7 |
| 2.1 Kestabilan Sistem Tenaga Listrik | 7 |
| 2.1.1 Kestabilan <i>Steady State</i> | 7 |
| 2.1.2 Kestabilan Transien | 8 |
| 2.1.3 Kestabilan Dinamik | 8 |
| 2.2 Permodelan Sistem Tenaga Listrik | 8 |
| 2.2.1 Permodelan Linier Mesin Sinkron..... | 8 |
| 2.2.2 Permodelan Governor | 12 |
| 2.2.3 Permodelan Sistem Eksitasi | 13 |
| 2.2.4 Permodelan Turbin dan Pengendaliannya..... | 14 |
| 2.2.5 Permodelan Sistem <i>Single Machine Infinite Bus</i> | 15 |
| 2.2.6 Permodelan <i>Power System Stabilizer</i> | 16 |
| 2.3 Reduksi Jaringan Menggunakan <i>Losses Concept</i> | 18 |

| | | |
|-------|---|----|
| 2.4 | <i>Particle Swarm Optimization</i> | 19 |
| 2.4.1 | Implementasi PSO..... | 21 |
| 2.5 | <i>Random Drift Particle Swarm Optimization</i> | 23 |
| 2.6 | Jenis Jenis Osilasi Gangguan | 25 |
| 2.6.1 | Mode Osilasi <i>Intraplant</i> | 25 |
| 2.6.2 | Mode Osilasi <i>local plant</i> | 25 |
| 2.6.3 | Mode Osilasi <i>Interarea</i> | 26 |
| 2.6.4 | Mode Osilasi Kontrol..... | 26 |
| 2.6.5 | Mode Osilasi Torsional | 26 |

BAB III PENALAN OPTIMAL PSS SMIB UNIT II PLTU

| | | |
|---------|---|----|
| | PACITAN PADA SISTEM 150 KV JAWA TIMUR | 27 |
| 3.1 | Alur penyelesaian Tugas Akhir | 27 |
| 3.2 | Permodelan Single Machine Infinite Bus Unit II PLTU Pacitan..... | 28 |
| 3.2.1 | Permodelan Mesin Sistem Eksitasi dan PSS | 28 |
| 3.2.2 | Konstanta K | 30 |
| 3.2.2.1 | Konstanta K1 dan K2 | 30 |
| 3.2.2.2 | Konstanta K3 dan K4 | 31 |
| 3.2.2.3 | Konstanta K5 dan K6 | 32 |
| 3.3 | Data Pembangkit, dan Saluran SMIB Unit II Pacitan pada Sistem Kelistrikan 150 kV Jawa Timur..... | 34 |
| 3.4 | <i>Statespace</i> dan <i>Eigenvalue</i> | 36 |
| 3.5 | Penghitungan Nilai Impedansi Ekvivalen dengan <i>Losses</i> <i>Concept</i> | 37 |
| 3.6 | Penalaan Parameter PSS Menggunakan RDPSO..... | 39 |

BAB IV SIMULASI DAN ANALISIS

| | | |
|-------|--|----|
| 4.1 | Analisis <i>Eigenvalue</i> | 43 |
| 4.2 | Analisis <i>Damping Ratio</i> | 46 |
| 4.3 | Analisis CDI..... | 48 |
| 4.4 | Analisis <i>Settling Time</i> dan <i>Overshoot</i> | 48 |
| 4.4.1 | Deviasi Sudut Rotor Akibat Perubahan Beban (0,01 p.u, dan 0,05 p.u) | 49 |
| 4.4.2 | Deviasi Frekuensi Akibat Perubahan Beban (0,01 p.u, dan 0,05 p.u)..... | 51 |
| 4.5 | Kurva Konvergensi dan Parameter PSS Hasil PSO | 53 |
| 4.6 | Perbandingan Performa RDPSO dan PSO | 54 |

| | |
|------------------------------|--|
| BAB V PENUTUP | 57 |
| 5.1 Kesimpulan | 57 |
| 5.2 Saran..... | 57 |
| DAFTAR PUSTAKA | 59 |
| INDEKS | 61 |
| LAMPIRAN I | Pemrograman RDPSO |
| LAMPIRAN II | Simulink <i>Exciter</i> PSS/E Model ESST4B |
| LAMPIRAN III | Simulink <i>Governor</i> Model IEEE G1 |
| LAMPIRAN IV | Penghitungan <i>Settling Time</i> dan <i>Overshoot</i> |
| LAMPIRAN V | Tentang PLTU Pacitan |
| LAMPIRAN VI | <i>Single Line Diagram</i> Sistem kelistrikan Jawa-Timur |
| RIWAYAT HIDUP PENULIS | |

DAFTAR TABEL

| | | |
|-------------------|--|----|
| Tabel 3.1 | Data Pembangkitan Unit II PLTU Pacitan..... | 34 |
| Tabel 3.2 | Parameter Dinamik Unit II PLTU Pacitan | 34 |
| Tabel 3.3 | Data Saluran Sistem 150 Kv Jawa Timur | 34 |
| Tabel 3.4 | Data Mesin | 34 |
| Tabel 3.5 | Data AVR..... | 34 |
| Tabel 3.6 | Data Governor dan Turbin | 35 |
| Tabel 3.7 | Data PSS Yang Belum Dioptimalkan | 35 |
| Tabel 3.8 | Data Aliran Daya Unit II PLTU Pacitan | 37 |
| Tabel 3.9 | Data Aliran Daya Sistem 150 Kv Jawa Timur | 37 |
| Tabel 3.10 | Data Aliran Daya (Semua Mesin Dimatikan) | 38 |
| Tabel 3.11 | Variabel Masukan Pada RDPSO | 40 |
| Tabel 3.12 | Interval Parameter Penalaan PSS..... | 41 |
| Tabel 4.1 | Nilai <i>Eigenvalue</i> Dari Sistem SMIB Tanpa PSS..... | 43 |
| Tabel 4.2 | <i>Eigenvalue</i> SMIB-PSS tidak Dioptimalkan | 44 |
| Tabel 4.3 | <i>Eigenvalue</i> dari sistem SMIB-PSS-RDPSO | 45 |
| Tabel 4.4 | <i>Damping Ratio</i> Berdasarkan <i>Eigenvalue</i> pada SMIB non PSS..... | 46 |
| Tabel 4.5 | <i>Damping Ratio</i> SMIB dengan PSS non RDPSO..... | 47 |
| Tabel 4.6 | <i>Damping Ratio</i> SMIB dengan PSS Tertala PSO | 47 |
| Tabel 4.7 | Perbandingan Nilai CDI..... | 48 |
| Tabel 4.8 | <i>Time Settling</i> dan <i>Overshoot</i> pada Deviasi Sudut Rotor..... | 51 |
| Tabel 4.9 | <i>Time Settling</i> dan <i>Overshoot</i> pada Deviasi Frekuensi | 53 |
| Tabel 4.10 | Parameter Optimal PSS Dengan Menggunakan PSO | 53 |
| Tabel 4.11 | Perbandingan Performa RDPSO vs PSO | 55 |

DAFTAR GAMBAR

| | |
|--|----|
| Gambar 2.1 Sistem Pembangkit Listrik Secara Umum..... | 9 |
| Gambar 2.2 Dinamika Dasar Generator Sinkron | 10 |
| Gambar 2.3 Diagram Blok Governor | 13 |
| Gambar 2.4 Diagram Blok Sistem Eksitasi | 13 |
| Gambar 2.5 Diagram Blok Turbin Uap dan Sistem Pengendaliannya | 14 |
| Gambar 2.6 Permodelan Linier dari Generator dan Sistem Eksitasi... | 15 |
| Gambar 2.7 Blok Diagram PSS | 17 |
| Gambar 2.8 <i>Single Machine Infinite Bus</i> | 18 |
| Gambar 3.1 <i>Flowchart</i> Pengerjaan Tugas Akhir | 27 |
| Gambar 3.2 Model Generator dan Sistem Eksitasi Unit II PLTU Pacitan dengan PSS | 29 |
| Gambar 3.3 Model PSS pada Unit II PLTU Pacitan | 30 |
| Gambar 3.4 Algoritma RDPSO pada Penalaan Optimal I..... | 39 |
| Gambar 3.5 Algoritma RDPSO pada Penalaan Optimal II | 40 |
| Gambar 4.1 Respon Sudut Rotor non-PSS pada Gangguan 0.01 p.u.. | 44 |
| Gambar 4.2 Deviasi Sudut Rotor Pada Perubahan Beban 0.01 p.u. ... | 49 |
| Gambar 4.3 Deviasi Sudut Rotor Pada Perubahan Beban 0.05 p.u. ... | 50 |
| Gambar 4.4 Deviasi Frekuensi Akibat Perubahan Beban 0.01 p.u. | 52 |
| Gambar 4.5 Deviasi Frekuensi Akibat Perubahan Beban 0.05 p.u. | 52 |
| Gambar 4.6 Grafik Konvergensi PSO | 54 |
| Gambar 4.7 Grafik Perbandingan Performa PSO vs RDPSO..... | 55 |

BAB I

PENDAHULUAN

1.1. Latar Belakang

Pembangkit Listrik Tenaga Uap (PLTU) Pacitan merupakan salah satu dari realisasi program percepatan pembangunan pembangkit listrik 10 MW di Indonesia. PLTU yang memproduksi daya berkekuatan 2×315 MW tersebut dibangun di Jalan Pacitan – Trenggalek Km-55, Desa Sukorejo, Kecamatan Sudimoro. Proyek ini bertujuan memenuhi kebutuhan listrik di Jawa Timur yang terus meningkat. Pembangunan pembangkit ini akan membantu pengembangan wilayah lintas selatan sehingga banyak investor yg akan masuk ke Jawa Timur dan akan mengurangi kemiskinan di Jawa Timur.

Sistem interkoneksi Jawa Timur 150 kV merupakan bagian dari sistem kelistrikan yang disuplai oleh PLTU Pacitan. Sistem ini merupakan sistem yang menanggung beban yang cukup besar. Sistem tenaga listrik dalam skala besar merupakan sistem yang rentan terhadap gangguan. Terjadinya suatu perubahan rasio permintaan beban dan total daya yang dihasilkan pada sistem ini dapat menyebabkan terjadinya osilasi pada sistem. Hal ini sejalan dengan hipotesis dari tim peneliti Pusat Pengaturan Jawa-Bali pada pasca gangguan di PLTU Pacitan pada tahun 2014. Osilasi yang terjadi pada pembangkit Pacitan diduga juga disebabkan oleh nilai redaman yang jelek dan pembangkit menyuplai beban yang jaraknya cukup jauh dari lokasi pembangkit.

Untuk mengatasi osilasi, ditempatkan peralatan pembantu untuk meredam osilasi. Salah satu peralatan yang digunakan adalah *Power System Stabilizer* (PSS) yang dipasang pada AVR [2]. Dengan menggunakan PSS, osilasi pada sistem dapat diredam sehingga tidak menyebabkan generator bergeser dari kondisi stabil. Peredaman osilasi pada sistem interkoneksi sangat penting untuk keamanan operasi sistem.

Dalam rangka melakukan analisis permasalahan diatas, pada Tugas Akhir ini, dilakukan kajian kestabilan dinamik PLTU Pacitan yang menyuplai sistem kelistrikan 150 kV Jawa-Timur. Pembahasan dikhususkan pada permasalahan penalaan PSS. *Zeq* (impedansi ekuivalen) yang digunakan untuk memodelkan *Single Machine Infinite Bus* sistem tersebut (dari sisi Pembangkit Pacitan) didapat dengan menggunakan metode *losses concept*. Metode *losses concept* adalah metode untuk mencari nilai *Zeq* terhadap satu mesin pembangkit dengan cara mencari

nilai rugi-rugi daya pada setiap bus yang ada di sistem tersebut dan hanya berasal dari pembangkit yang sedang diamati (*observed machine*) [10]. Proses optimisasi penalaan parameter PSS dilakukan dengan algoritma RDPSO (*Random Drift Particle Swarm Optimization*). Dipublikasikan pada tahun 2015, RDPSO adalah pengembangan baru metode *Particle Swarm Optimization* (PSO) yang ditemukan pada tahun 1995, metode ini mengadopsi pergerakan elektron pada sebuah konduktor metal. Perpindahan elektron pada konduktor metal adalah superposisi dari perpindahan yang disebabkan oleh suhu lingkungan, dan medan listrik itu sendiri. Resultan dari pergerakan elektron akan berpindah dari tempat dengan potensial tinggi ke potensial rendah.

Fungsi objektif RDPSO menggunakan *Comprehensive Damping Index* (CDI). Nilai respon diketahui dengan menganalisis nilai *overshoot* dan *settling time* [6]. Kemudian, hasil simulasi dianalisis dengan cara membandingkan hasil simulasi sistem menggunakan PSS, dan sistem menggunakan PSS yang dioptimalkan.

1.2. Tujuan

Tugas Akhir ini bertujuan untuk mendapatkan nilai impedansi ekuivalen dengan menggunakan *losses concept*, dan kemudian mencari nilai yang optimal untuk penalaan PSS menggunakan RDPSO, pada model sistem kelistrikan SMIB (*Single Machine Infinite Bus*) Pembangkit Pacitan dengan permodelan generator klasik, sehingga pada akhirnya, akan didapat perbandingan respon kestabilan dinamik sistem, antara sistem tanpa PSS, dengan PSS yang belum dioptimalkan, dan dengan penalaan PSS yang telah dioptimalkan.

1.3. Perumusan Masalah

Adapun fokus permasalahan yang terdapat pada buku ini adalah sebagai berikut.

1. Permodelan *Power System Stabilizer* (PSS) dan penerapannya pada generator dengan sistem *Single Machine Infinite Bus* (SMIB).
2. Penghitungan nilai impedansi ekuivalen sistem dari sisi pembangkit Pacitan menggunakan metode *losses concept*. Nilai ini akan digunakan untuk memodelakan SMIB pembangkit Pacitan.

3. Penentuan koordinasi parameter PSS yang optimal pada pembangkit Pacitan dengan menggunakan RDPSO (*Random Drift Particle Swarm Optimization*)
4. Pengamatan respon kestabilan dinamik sistem (sudut rotor, dan frekuensi), antara sistem tanpa PSS, dengan PSS yang belum dioptimalkan, dan sistem dengan PSS yang dioptimalkan dengan RDPSO.

1.4. Batasan Masalah

Terdapat batasan masalah pada Tugas Akhir ini, antara lain:

1. Metode yang dipakai untuk menyelesaikan permasalahan adalah metode RDPSO, dengan menggunakan PSS jenis konvensional, dan disimulasikan dengan MATLAB *Simulink*.
2. Analisis sistem dilakukan dengan permodelan *Single Machine Infinite Bus* (SMIB)
3. Beban pada sistem dianggap statis, faktor harmonisa diabaikan, dan faktor ekonomis tidak diperhitungkan (Permodelan generator Klasik), atau biasa disebut permodelan generator dan sistem eksitasi *Heffron Phillips* (IAEE tahun 1957, dengan model turbin uap *high pressure*).
4. Penentuan *Req* dan *Xeq* dilakukan dengan metode *losses concept*.
5. PSS pada Tugas akhir ini dibatasi hanya untuk mengatasi gangguan *local area mode oscillations (Local Disturbance)*, dan berupa *small disturbance*.
6. Analisis COS tidak dilakukan.

1.5. Metode

Metode yang digunakan pada Tugas Akhir ini terbagi menjadi tiga tahap, yaitu tahap permodelan sistem, tahap pengujian sistem, dan tahap analisis. Adapun keterangan mengenai masing-masing tahapan yang dilakukan dalam Tugas Akhir ini adalah sebagai berikut:

1.5.1. Tahapan Permodelan Sistem

Adapun langkah-langkah pada tahap permodelan sistem antara lain:

a. Pencarian Impedansi Ekvivalen Sistem

Tahap ini dilakukan untuk menentukan nilai R_{eq} dan X_{eq} sistem apabila dipantau dari sisi Pembangkit Pacitan dengan menggunakan *losses concept*.

b. Permodelan Generator dan Sistem Eksitasi

Pada tahap ini, akan dilakukan permodelan generator dan sistem eksitasi *Heffron Phillips* dengan menggunakan MATLAB *Simulink*.

c. Permodelan Kontrol PSS

Pada tahap ini, didalam *Simulink file* yang sama, dibuat model generator dan sistem eksitasi yang sama, namun dengan tambahan model kontrol PSS. Kontrol PSS yang dipilih adalah permodelan kontrol PSS Konvensional dengan enam variabel sebagai objek optimasi metode RDPSO: K_{pss} , T_w , T_1 , T_2 , T_3 , dan T_4 .

d. Penghitungan Nilai K

Pada tahap ini, akan dilakukan penghitungan nilai K (K_1 , K_2 , K_3 , K_4 , K_5 , K_6) pada model generator dan sistem eksitasi *Heffron Phillips*. Penghitungan ini melibatkan data tegangan output *steady state* (sumbu q dan d), arus stator *steady state*, sudut daya rotor *steady state*, tegangan bus tak hingga, serta tegangan transien.

1.5.2. Pengujian Sistem

Pada tahap ini, dilakukan penambahan kontrol PSS pada permodelan generator dan sistem eksitasi.

1.5.3. Tahap Optimasi Parameter Kontrol PSS

Setelah melakukan reduksi jaringan, memodelkan generator, sistem eksitasi, dan kontrol PSS, akan dilakukan *tunning* PSS dengan metode RDPSO untuk mendapatkan *overshoot* dan *settling time* yang baik pada respon frekuensi dan sudut rotornya. Fungsi objektif yang digunakan pada RDPSO adalah *Comprehensive Damping Index* (CDI).

1.5.4. Tahap Analisis

Simulasi dilakukan pada sistem yang menggunakan PSS dan sistem menggunakan PSS yang telah dioptimalkan penalaannya pada masing masing kasus. Hasil simulasi dilakukan dengan menganalisis data dengan membandingkan hasil simulasi sistem. Penulisan laporan dilakukan sebagai penggambaran dan pendiskripsian kesimpulan dari Tugas Akhir ini. Kesimpulan kemudian disajikan dalam bentuk hasil analisis dan perbandingan. Selain itu saran untuk keberlanjutan penelitian akan diberikan.

1.6. Sistematika Penulisan

Tugas Akhir ini dibagi menjadi lima bab. Adapun susunannya adalah sebagai berikut:

BAB I. Pendahuluan

Bab ini berisi tentang latar belakang, permasalahan, tujuan, metode penelitian, dan sistematika penulisan serta relevansi atau manfaat laporan.

BAB II. Dasar Teori

Bab ini berisi teori-teori penunjang yang berhubungan dengan permasalahan yang dibahas pada Tugas Akhir ini. Teori ini digunakan sebagai dasar dalam penulisan buku ini.

BAB III. Penentuan Nilai Impedansi Ekuivalen Dengan *Losses Concept* dan Optimisasi Penalaan PSS

Bab ini berisi tentang penjelasan mengenai penerapan PSS yang ditala dengan *Random Drift Particle Swarm Optimization* beserta penentuan Z_{eq} dengan *losses concept*.

BAB IV. Simulasi dan Analisis

Bab ini berisi tentang pengujian sistem dengan PSS yang belum dioptimalkan, dan dengan PSS yang telah dioptimalkan.

BAB V. Penutup

Bab ini berisi tentang penutup yang memuat tentang kesimpulan yang dapat diambil dari hasil pengujian dan perbandingan yang telah dilakukan.

1.7. Relevansi atau Manfaat

Buku ini diharapkan dapat memberikan kontribusi terhadap keilmuan di bidang sistem kelistrikan, khususnya permasalahan penalaan PSS dalam rangka meningkatkan kualitas kestabilan dinamik pada sistem kelistrikan. Pada masa yang akan datang, buku ini dapat dijadikan sebagai referensi untuk mahasiswa maupun praktisi untuk melakukan studi analisis kestabilan dinamik khususnya pada permasalahan penalaan PSS di pembangkit manapun yang tersambung dengan *grid* di Indonesia.

BAB II

DASAR TEORI

2.1 Kestabilan Sistem Tenaga Listrik

Dalam sebuah sistem operasi, kestabilan sistem menjadi perhatian utama. Kestabilan sistem tenaga listrik adalah kemampuan suatu sistem tenaga listrik atau bagian komponennya untuk mempertahankan sinkronisasi dan keseimbangan jika terjadi gangguan. Kondisi yang tidak stabil ditunjukkan dengan tegangannya osilasi dan keseimbangan pada sistem.

Terdapat keseimbangan antara daya input mekanis (*prime mover*) dengan daya output elektrik (beban listrik) pada sistem tenaga listrik yang stabil. Pada keadaan ini, generator berputar pada kecepatan sinkron. Kenaikan atau penurunan beban harus diikuti dengan perubahan daya input mekanis dari *prime mover*. Jika daya input mekanik tidak dapat menyesuaikan dengan perubahan daya output elektrik dan rugi-rugi sistem, maka ketidakstabilan pada sistem akan terjadi. Kecepatan rotor generator (frekuensi sistem) dan tegangan akan menyimpang dari kondisi normal. Ketika terjadi gangguan, sesaat akan terjadi peredaman besar antara daya input dari *prime mover* dengan daya output dari generator. Putaran rotor pada generator akan mengalami percepatan atau pelambatan. Jika tidak segera dihilangkan, maka percepatan atau pelambatan rotor generator ini akan menyebabkan hilangnya kesinkronan sistem.

Berdasarkan sifat dan besar gangguan, kestabilan dalam sistem tenaga listrik terbagi menjadi tiga: kestabilan *steady-state*, kestabilan transien, dan kestabilan dinamik [3].

2.1.1 Kestabilan Steady State

Kestabilan *steady state* adalah kemampuan sistem tenaga untuk mencapai kondisi stabil pada kondisi operasi baru yang sama atau identik dengan kondisi sebelum terjadi gangguan setelah sistem mengalami gangguan kecil. Analisis kestabilan *steady state* menggunakan pendekatan model linear. Kestabilan *steady state* pada sistem tenaga dapat disebut sebagai kestabilan sinyal kecil (*small signal stability*). Kestabilan *steady state* merupakan sebuah fungsi dari kondisi operasi [1].

2.1.2 Kestabilan Transien

Kestabilan transien adalah kemampuan sistem tenaga untuk mencapai kondisi stabil operasi baru yang dapat diterima setelah sistem mengalami gangguan besar. Analisis kestabilan transien menggunakan pendekatan model nonlinear. Kestabilan transien pada sistem tenaga adalah respon output yang mencapai kondisi operasi steady-state yang diizinkan dan sistem yang dapat kembali ke posisi semula pada saat sistem mengalami gangguan. Kestabilan transien merupakan fungsi dari kondisi operasi dan gangguan [1].

2.1.3 Kestabilan Dinamik

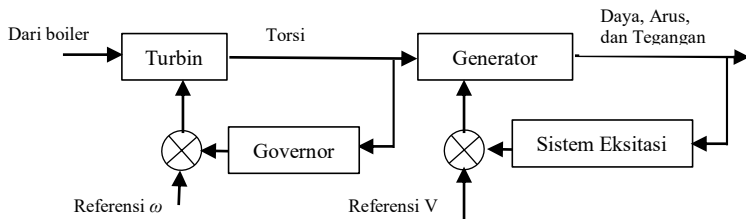
Kestabilan dinamik sistem tenaga listrik adalah kestabilan sistem tenaga listrik akibat adanya perubahan beban yang sangat kecil. [1]. Kestabilan dinamik ini sama dengan kestabilan *steady state* namun yang membedakan yaitu sistem eksitasi, turbin, dan generator dimodelkan dengan memperhitungkan variasi medan magnet pada air gap mesin, sedangkan pada kestabilan *steady state* generator direpresentasikan sebagai sumber tegangan yang konstan atau tidak terjadi variasi medan magnet. Sumber lain menyebutkan bahwa kestabilan ini berarti kemampuan sistem untuk tetap mempertahankan sinkronisasi setelah initial swing atau periode transien hingga pada kondisi titik keseimbangan *steady state* [3].

2.2 Pemodelan Sistem Tenaga Listrik

Kesahihan analisis studi kestabilan dinamik (kestabilan di daerah sekitar titik kerja) jaringan tenaga listrik yang meliputi respon dinamik sistem tergantung pada kesahihan pemodelan sistem tersebut [1]. Untuk itu perlu memahami model dinamik sistem tenaga listrik yang meliputi generator sinkron, sistem eksitasi dan sistem governor turbin.

2.2.1 Pemodelan Linear Mesin Sinkron

Secara umum sistem pembangkit dapat digambarkan seperti pada Gambar 2.1 [1]



Gambar 2.1 Sistem Pembangkit Listrik Secara Umum

Governor merupakan bagian dari unit pembangkit yang berfungsi untuk mengatur pemberian bahan bakar (uap atau air) pada sistem pembangkitan agar didapatkan kecepatan rotor yang stabil. Bila ada perubahan pada terminal keluaran generator akibat perubahan beban, maka akan terjadi fungsi *feedback* yang diatur oleh governor untuk menyesuaikan kembali putaran rotor.

Sistem eksitasi adalah sistem pengendali keluaran generator, seperti tegangan, arus dan faktor daya. Bila ada perubahan pada keluaran generator tersebut, maka sistem eksitasi yang berfungsi untuk mengendalikan generator agar menyesuaikan dan mencari titik keseimbangan baru.

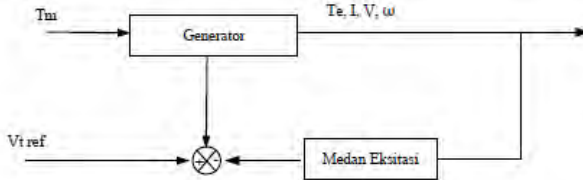
Kedua peralatan sistem kontrol (governor dan sistem eksitasi) mempunyai perbedaan waktu respon. Governor mempunyai respon yang lambat terhadap perubahan beban, sedangkan sistem eksitasi responnya lebih cepat. Namun, karena governor mempunyai waktu respon yang lebih lambat, maka dalam studi kestabilan dinamik, yang diperhatikan adalah sistem eksitasi.

Seperti telah dijelaskan, bahwa dengan adanya perubahan beban atau gangguan kecil, akan berimbas terhadap perubahan kecepatan rotor. Perubahan kecepatan rotor ini secara langsung akan berakibat terhadap perubahan kecepatan sudut rotor sehingga berimbas pada keluaran generator. Jika kecepatan sudut rotor akan berosilasi di sekitar nilai sinkronnya, maka tegangan keluarannya akan berosilasi di sekitar harga nominalnya.

Maka dapat diambil kesimpulan bahwa kestabilan dinamik didapat dari adanya perubahan beban kecil atau gangguan yang mengakibatkan perubahan kecepatan sudut rotor dan tegangan keluaran namun sistem pembangkitannya yaitu governor dan sistem eksitasi mampu

menyesuaikannya dan menemukan titik keseimbangan baru [4].

Dinamika dasar dari sebuah mesin sinkron diperlihatkan pada Gambar 2.2.



Gambar 2.2 Dinamika Dasar Generator Sinkron

Suatu generator agar menghasilkan energi listrik, membutuhkan dua masukan, yang pertama torsi mekanik turbin (T_m) dan yang kedua fluksi medan magnet (E_{FD}). Torsi mekanik berfungsi sebagai pemutar generator sinkron. Fluksi medan magnet dihasilkan dari rangkaian medan melalui lilitan yang terdapat pada rotor. Dengan adanya torsi mekanik, rotor generator berputar dengan energi kinetik $\frac{1}{2} J \omega^2$ Joule. Momentum sudut

$$M = J \omega \quad (2.1)$$

Dengan ω (radian per detik) merupakan kecepatan sudut dan J adalah momen inersia (kg.m^2). Fluksi yang dibangkitkan oleh kumparan medan rotor dengan arus I_f akan berputar dan menginduksikan gaya gerak listrik pada kumparan stator sebesar

$$E = c n \phi \quad (2.2)$$

Jika generator sinkron dibebani, maka akan mengalir arus generator ke beban. Arus ini menghasilkan fluksi pada stator dan akan menimbulkan torsi elektrik (T_e) melawan torsi mekanik. Pada kondisi tunak, jumlah torsi mekanik dengan torsi elektrik sama dengan nol ($T_m - T_e = 0$), dan generator akan berputar pada kecepatan sudut sinkron (ω_0). Sebelum mencapai kondisi tunak ada masa transien dan jumlah torsi ini menimbulkan torsi akselerasi dan akan menghasilkan percepatan atau perlambatan, persamaan gerak pada kondisi ini adalah:

$$T a = J \frac{d^2 \delta m}{dt} = T_m - T_e \quad (2.3)$$

Dengan

$$T_m = \text{Torka mekanik turbin (N.m)}$$

T_e = Torka elektrik rotor (N.m)
 T_a = Percepatan torka (N.m)
 T = waktu (detik)
 δm = sudut mekanis rotor (rad-mekanik)
 J = total momen inersia mesin sinkron (kg.m²)

Persamaan (2.3) dikali dengan kecepatan sudut (ω_m), sehingga didapat persamaan :

$$\omega_m J \frac{d^2 \delta m}{dt} = P_a = P_m - P_e \quad (2.4)$$

Dengan

P_m = Daya mekanik pada rotor (Watt)
 P_e = daya elektrik pada rotor (Watt)
 P_a = daya percepatan (Watt)
 ω_m = Kecepatan sudut sinkron (rad mekanik/s)

Persamaan diatas dapat diubah menjadi

$$\frac{2}{\omega_m} \left(\frac{1}{2} \omega_m^2 J \right) \frac{d^2 \delta m}{dt} = P_m - P_e \quad (2.5)$$

Jika persamaan (2.5) dibagi dengan rating MVA generator (S), maka:

$$\frac{2}{\omega_m S} \left(\frac{1}{2} \omega_m^2 J \right) \frac{d^2 \delta m}{dt^2} = P_m - P_e \quad (2.6)$$

Jika

$$H = \frac{\text{Energi kinetik pada kecepatan sinkron}}{\text{Rating MVA}}$$

$$H = \left(\frac{1}{2} \omega_m^2 J \right) / S \quad (2.7)$$

Maka persamaan (2.6) dapat disederhanakan menjadi

$$\frac{2H}{\omega_m} \frac{d^2 \delta m}{dt^2} = P_m - P_e(p.u) \quad (2.8)$$

Dengan

$$\omega_m = \frac{2}{p} \omega_s \quad (2.9)$$

$$\delta_m = \frac{2}{p} \delta \quad (2.10)$$

ω_s = kecepatan sudut sinkron (rad/s)

δ = sudut beban / elektrik rotor (rad)
 p = jumlah kutub generator

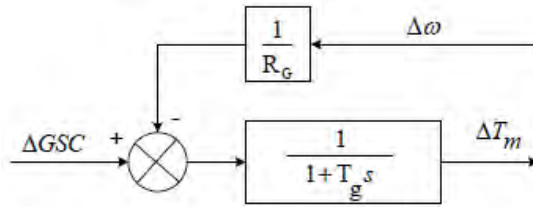
Persamaan (2.8) disebut dengan persamaan ayunan mesin yang merupakan persamaan dasar yang mengatur dinamika (gerak) perputaran mesin sinkron dalam studi kestabilan. Dari persamaan tersebut dapat diketahui bahwa perbedaan antara daya mekanik turbin dan daya elektrik generator menyebabkan sudut rotor mengalami percepatan atau perlambatan. Oleh karena mesin memiliki komponen torsi yang sebanding dengan kecepatan sudut, maka persamaan gerak rotor secara lengkap dinyatakan sebagai berikut:

$$\frac{M}{\omega_0} \frac{d^2 \delta m}{dt^2} + \frac{D}{\omega_0} \frac{d \delta m}{dt} = T_m - T_e \quad (2.11)$$

Dimana D merupakan koefisien redaman mesin. Suatu generator yang beroperasi pada kondisi tunak ($T_m = T_e$), kemudian terjadi penambahan beban kecil yang menyebabkan kenaikan fluksi medan stator dan perubahan E'_q . Perubahan kecil juga menyebabkan perubahan sudut mesin dan perubahan torsi elektrik, torsi elektrik akan naik dengan bertambahnya sudut mesin δ , dan E'_q .

2.2.2 Pemodelan Governor

Governor merupakan pengendali yang berfungsi untuk mengatur nilai torsi mekanik yang menjadi masukan generator. Besar perubahan torsi mekanik T_m tergantung pada konstanta *speed droop*, *transfer function* governor dan sumber energi. Perubahan nilai T_m dihasilkan oleh perubahan kecepatan ω , perubahan beban dan *speed reference* (*Governor Speed Changer*, GSC). Jika terjadi perubahan putaran rotor generator, maka governor akan memberi umpan balik untuk mencapai keseimbangan baru. Representasi governor ditunjukkan pada Gambar 2.7. Berdasarkan Gambar 2.3, perubahan kecepatan rotor generator $\Delta\omega$ atau ΔGSC dapat menghasilkan sebuah perubahan torsi mekanik mesin ΔT_m .



Gambar 2.3 Diagram Blok Governor

Dalam model ini, diasumsikan ΔGSC adalah menggunakan sinyal unit step dan efek penggabungan sistem turbin dengan *speed governor* menghasilkan daya mekanik P_m yang dirumuskan sebagai berikut.

$$P_m = -[K_g / (1 + T_g s)] \Delta \omega \quad (2.12)$$

Dengan,

K_g : Konstanta gain = $1/R_g$

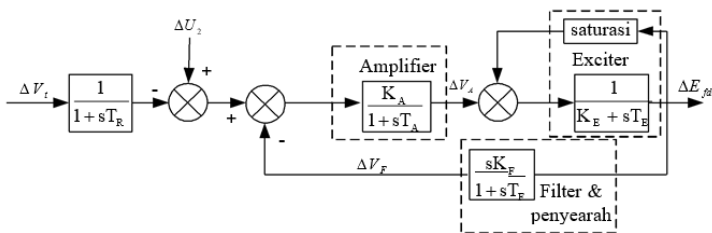
T_g : *Governor time constant*

R_g : Konstanta droop governor

ΔGSC : Perubahan acuan kecepatan (*Governor Speed Changer*)

2.2.3 Pemodelan Sistem Eksitasi

Model sistem eksitasi dalam pengaturan tegangan mengacu pada model IEEE tipe 1 [1] seperti yang ditunjukkan pada Gambar 2.4



Gambar 2.4 Diagram Blok Sistem Eksitasi

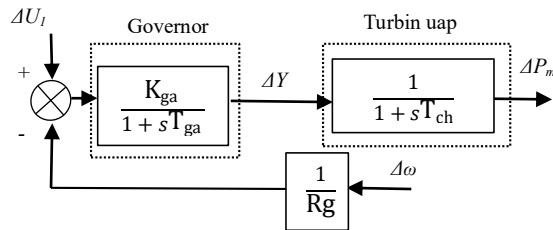
Parameter sistem ekstasi di atas dapat dijelaskan sebagai berikut.

- K_A : konstanta penguatan amplifier
- K_F : konstanta penguatan filter
- K_E : konstanta penguatan exciter
- T_A : waktu tanggap amplifier
- T_E : waktu tanggap exciter
- T_F : waktu tanggap filter
- ΔU_2 : perubahan sinyal kontrol mesin

Harga konstanta T_R biasanya kecil sekali, sehingga dapat diabaikan. Dengan ΔV_A = perubahan tegangan amplifier dan ΔV_F = perubahan tegangan keluaran penyearah.

2.2.4 Pemodelan Turbin dan Sistem Pengendaliannya

Model turbin dan sistem pengendaliannya yang dipakai disini adalah model turbin uap dan pengendaliannya, mengaju pada model standar IEEE [1]. Turbin uap memiliki input berupa energi mekanik yang disalurkan dari ketel uap dan memiliki output energi (torka) mekanik yang digunakan untuk menggerakkan turbin uap. Model turbin uap dalam bentuk linier dapat dilihat pada Gambar 2.5



Gambar 2.5 Diagram Blok Turbin Uap dan Sistem Pengendaliannya

Parameter turbin uap diatas dapat dijelaskan sebagai berikut:

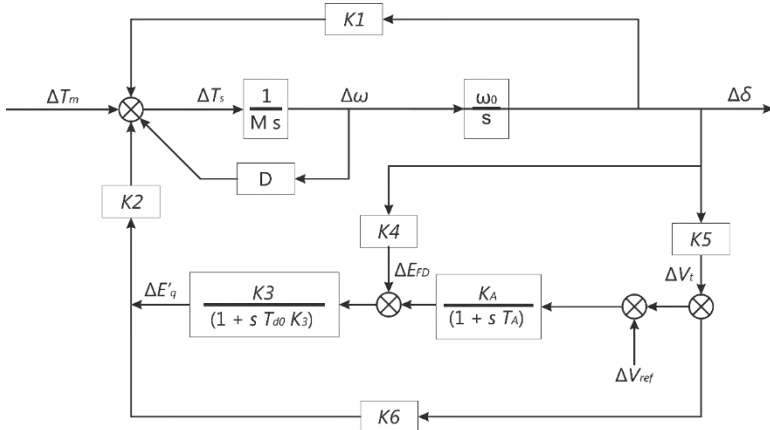
- ΔY : perubahan tinggi katup
- T_w : waktu tanggap turbin uap
- T_{ga} : waktu tanggap pengatur turbin uap
- T_{ch} : konstanta waktu turbin uap (s)
- K_{ga} : penguatan pengatur turbin uap
- R_g : konstanta pengatur turbin uap
- ΔU_1 : perubahan sinyal kontrol umpan balik

ΔP_m : perubahan daya mekanik

2.2.5 Pemodelan Sistem *Single Machine Infinite Bus*

Dari perumusan blok diagram yang dijabarkan diatas, maka dapat dibentuk model linier generator dan sistem eksitasi. Blok diagram permodelan Turbin, governor, dan sistem eksitasi disesuaikan dengan penjabaran yang telah dijelaskan sebelumnya. Gambar 2.6 menyatakan permodelan linier dari generator dan sistem eksitasi.

Pada umumnya transmisi pada sistem tenaga didesain terhubung secara interkoneksi, yang memungkinkan banyak mesin (multimesin) untuk menyuplai permintaan beban secara bersama-sama. Pemilihan model analisis kestabilan dinamik dapat disesuaikan dengan kebutuhan analisis. Pada Tugas Akhir ini, model SMIB dipilih untuk melakukan analisis osilasi mesin. Oleh karena pemodelan SMIB hanya fokus mengamati satu mesin saja yang terhubung dengan bus infinite melalui impedansi ekuivalen, sehingga perlu adanya reduksi jaringan untuk mendapatkan impedansi ekuivalen. Banyak metode reduksi jaringan yang telah dikembangkan, salah satu dari reduksi jaringan yang sering digunakan adalah Rei-Dimo. Namun Rei-Dimo digunakan untuk mengamati *multi-machines*.



Gambar 2.6 Permodelan Linier dari Generator dan Sistem Eksitasi

2.2.6 Pemodelan *Power System Stabilizer*

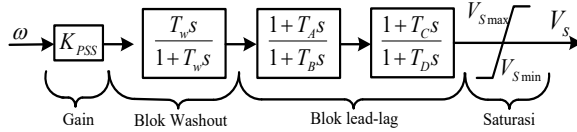
PSS banyak digunakan dalam sistem tenaga listrik untuk memperbaiki kestabilan dinamik, PSS digunakan sebagai kontroler sistem eksitasi untuk menambah redaman pada osilasi rotor. Untuk menghasilkan komponen redaman PSS memproduksi komponen torsi elektrik yang sesuai dengan deviasi pada kecepatan rotor. PSS harus ditala dengan tepat, untuk membantu exciter dalam meredam osilasi.

PSS menerima input berupa perubahan kecepatan rotor untuk menghasilkan sinyal tambahan sebagai kontroler exciter. Exciter mempengaruhi besar tegangan medan yang dihasilkan pada sisi rotor dan mempengaruhi besar fluks magnetik yang dibangkitkan. Fluks magnetik berbanding lurus dengan besar torka elektrik yang dihasilkan pada mesin. Torka elektrik melawan besar torka mekanik untuk meredam osilasi frekuensi yang terjadi pada mesin [3].

Agar dapat berfungsi dengan baik, PSS harus ditala dengan tepat. Metode desain PSS secara umum melibatkan frekuensi respon yang berdasarkan pada konsep peningkatan redaman torsi. Transfer function PSS ditala untuk menyediakan karakteristik *phase-lead* yang tepat untuk mengkompensasi *phase-lag* antara frekuensi input automatic voltage regulator Δv_s dan torsi elektrik [3]. Sehingga, komponen torsi elektrik sefasa dengan variasi kecepatan untuk memperbaiki redaman. Dengan menggunakan model matematika PSS yang sederhana, sehingga model matematika PSS dapat ditulis

$$V_s = K_{pss} \frac{T_w s}{1 + T_w s} \left[\frac{(1 + sT_A)(1 + sT_C)}{(1 + sT_B)(1 + sT_D)} \right] \omega \quad (2.13)$$

Dengan mengasumsikan bahwa output dari PSS adalah V_s dengan input $\Delta \omega$, maka Persamaan 2.13 dapat ditulis dalam diagram blok pada Gambar 2.7.



Gambar 2.7 Blok Diagram PSS

Blok diagram PSS pada Gambar 2.7 terdiri dari satu blok *gain*, satu blok *washout*, 2 blok *lead-lag*, dan *limiter*. Fungsi dari masing-masing blok adalah sebagai berikut :

a. Blok Gain

Gain berfungsi untuk mengatur besar penguatan agar diperoleh besaran torsi sesuai dengan yang diinginkan. Sinyal input PSS dapat diambil dari berbagai macam sinyal seperti perubahan rotor, keluaran daya listrik atau frekuensi terminal bus.

b. Blok Washout Filter

Wash out filter berfungsi untuk menyediakan bias *steady state* output PSS yang akan memodifikasi tegangan terminal generator. PSS diharapkan hanya dapat merespon variasi *transient* dari sinyal kecepatan rotor generator dan tidak untuk sinyal DC offset. *Wash out filter* bekerja sebagai *high pass filter* yang akan melewatkan semua frekuensi yang diinginkan. Apabila hanya mode lokal yang diinginkan, nilai T_w dapat dipilih dalam range 1 sampai 2. Tetapi, jika mode interarea juga ingin diredam, maka nilai T_w harus dipilih dalam interval 10 sampai 20. Nilai T_w yang lebih tinggi dapat memperbaiki respon tegangan sistem selama *island operation*.

c. Blok Lead-Lag

Lead-Lag berfungsi sebagai penghasil karakteristik *phase-lead* yang sesuai untuk mengkompensasi *phase-lag* antara masukan eksitasi dan torsi generator.

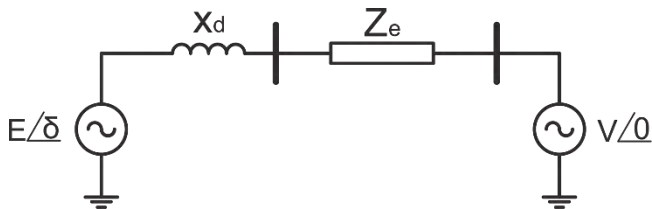
d. Limiter

Output PSS dibatasi agar aksi PSS pada AVR sesuai dengan yang diharapkan. Sebagai contoh, pada saat terjadi pelepasan beban, AVR beraksi untuk mengurangi tegangan terminal generator pada saat PSS menghasilkan sinyal kontrol untuk menaikkan tegangan (karena

kecepatan rotor generator bertambah besar pada saat terjadi pelepasan beban). Pada kondisi ini sangat diperlukan untuk menonaktifkan PSS. Hal ini menunjukkan pentingnya pembatasan nilai sinyal output PSS yang dapat dilakukan oleh blok *limiter*. Perlu diperhatikan bahwa, nilai Batasan negatif yang tinggi dapat mengganggu kestabilan swing yang pertama.

2.3 Reduksi Jaringan Menggunakan *Losses Concept*

Pada umumnya transmisi pada sistem tenaga didesain terhubung secara interkoneksi, yang memungkinkan banyak mesin (*multimesin*) menyuplai permintaan beban secara bersama-sama. Pemilihan model analisis kestabilan dinamik dapat disesuaikan dengan kebutuhan analisis, pada paper ini diambil model SMIB untuk analisis osilasi mesin. Karena pemodelan SMIB hanya fokus mengamati satu mesin saja yang terhubung dengan *infinite bus* melalui impedansi ekuivalen, sehingga perlu adanya reduksi jaringan untuk mendapatkan impedansi ekuivalen seperti pada gambar 2.8.



Gambar 2.8 *Single Machine Infinite Bus*

Pada umumnya metode reduksi jaringan yang digunakan adalah Rei-Dimo. Metode tersebut menyajikan perhitungan yang rumit dan digunakan untuk reduksi *multimesin*. Pada paper ini menyajikan metode reduksi jaringan untuk mendapatkan R_{eq} dan X_{eq} untuk mengamati model SMIB menggunakan konsep rugi-rugi yang lebih sederhana. Secara alami dalam sistem, daya disalurkan melalui jaringan transmisi yang memiliki losses. Berdasarkan *Losses Concept* tersebut R_{eq} dan X_{eq} untuk single mesin dapat dikalkulasi dengan lebih sederhana. Proses untuk reduksi jaringan menggunakan losses konsep setiap unit mesin dapat dijelaskan sebagai berikut :

1. Lakukan aliran daya sistem interkoneksi dengan banyak mesin (multimesin). Perhatikan nilai pembebanan pada mesin yang akan diamati.
2. Siapkan nilai data dengan mengatur pembebanan mesin yang diamati sehingga tidak melebihi kapasitas mula-mula dari hasil aliran daya sistem interkoneksi. Hal ini berguna untuk mendekati kondisi operasi mesin yang diamati.
3. Lakukan aliran daya dengan semua mesin tidak menyala, kecuali mesin yang diamati. Dapatkan daya aktif (P), daya reaktif (Q), losses daya aktif (P_L), dan losses daya reaktif (Q_L).
4. Hitung nilai arus mesin yang diamati menggunakan persamaan :

$$I = \frac{P-jQ}{\sqrt{3} \cdot V_i} \quad (2.14)$$

Dengan I = Arus mesin yang diamati.

$P - jQ$ = Daya kompleks yang dibangkitkan mesin yang diamati.

V_i = Tegangan konjugat kompleks pada bus mesin yang diamati.

5. Hitung nilai Req dan Xeq menggunakan konsep losses sesuai dengan persamaan :

$$Zeq = Req + jXeq = \frac{P_L + jQ_L}{|I|^2} \quad (2.15)$$

Dengan : P_L = Losses daya aktif.

Q_L = Losses daya reaktif.

Zeq = Impedansi ekuivalen saluran.

2.4 Particle Swarm Optimization

Particle Swarm Optimization, disingkat sebagai PSO, didasarkan pada perilaku sebuah kawanan serangga, seperti semut, rayap, lebah atau burung. Algoritma PSO meniru perilaku sosial organisme ini. Perilaku sosial terdiri dari tindakan individu dan pengaruh dari individu-individu lain dalam suatu kelompok. Kata partikel menunjukkan, misalnya, seekor burung dalam kawanan burung. Setiap individu atau partikel berperilaku secara terdistribusi dengan cara menggunakan *kecerdasannya* (intelligence) sendiri dan juga dipengaruhi perilaku kelompok kolektifnya. Dengan demikian, jika satu partikel atau seekor burung

menemukan jalan yang tepat atau pendek menuju ke sumber makanan, sisa kelompok yang lain juga akan dapat segera mengikuti jalan tersebut meskipun lokasi mereka jauh di kelompok tersebut. Metode optimasi yang didasarkan pada *swarm intelligence* ini disebut algoritma *behaviorally inspired* sebagai alternatif dari algoritma genetika, yang sering disebut *evolution-based procedures*.

Dalam konteks optimasi multivariabel, kawanan diasumsikan mempunyai ukuran tertentu atau tetap dengan setiap partikel posisi awalnya terletak di suatu lokasi yang acak dalam ruang multidimensi. Setiap partikel diasumsikan memiliki dua karakteristik: posisi dan kecepatan. Setiap partikel bergerak dalam *ruang/space* tertentu dan mengingat posisi terbaik yang pernah dilalui atau ditemukan terhadap sumber makanan atau nilai fungsi objektif. Setiap partikel menyampaikan informasi atau posisi bagusnya kepada partikel yang lain dan menyesuaikan posisi dan kecepatan masing-masing berdasarkan informasi yang diterima mengenai posisi yang bagus tersebut. Sebagai contoh, misalnya perilaku burung-burung dalam kawanan burung. Meskipun setiap burung mempunyai keterbatasan dalam hal kecerdasan, biasanya ia akan mengikuti kebiasaan (*rule*) seperti berikut :

1. Seekor burung tidak berada terlalu dekat dengan burung yang lain
2. Burung tersebut akan mengarahkan terbangnya ke arah rata-rata keseluruhan burung
3. Akan memposisikan diri dengan rata-rata posisi burung yang lain dengan menjaga sehingga jarak antar burung dalam kawanan itu tidak terlalu jauh

Dengan demikian perilaku kawanan burung akan didasarkan pada kombinasi dari tiga faktor simpel berikut:

1. Kohesi - terbang bersama
2. Separasi - jangan terlalu dekat
3. Penyesuaian(alignment) - mengikuti arah bersama

Jadi PSO dikembangkan dengan berdasarkan pada model berikut:

1. Ketika seekor burung mendekati target atau makanan (atau bisa minimum atau maximum suatu fungsi tujuan)

secara cepat mengirim informasi kepada burung-burung yang lain dalam kawanan tertentu

2. Burung yang lain akan mengikuti arah menuju ke makanan tetapi tidak secara langsung
3. Ada komponen yang tergantung pada pikiran setiap burung, yaitu memorinya tentang apa yang sudah dilewati pada waktu sebelumnya. Model ini akan disimulasikan dalam ruang dengan dimensi tertentu dengan sejumlah iterasi sehingga di setiap iterasi, posisi partikel akan semakin mengarah ke target yang dituju (minimasi atau maksimasi fungsi). Ini dilakukan hingga maksimum iterasi dicapai atau bisa juga digunakan kriteria penghentian yang lain.

2.4.1. Implementasi PSO

Misalkan kita mempunyai fungsi berikut :

$$\min f(x) \quad (2.16)$$

$$\text{dimana } X(B) \leq X \leq X(A) \quad (2.17)$$

$X(B)$ adalah batas bawah dan $X(A)$ adalah batas atas dari X . Prosedur algoritma PSO dapat dijabarkan dengan langkah – langkah sebagai berikut :

1. Asumsikan bahwa ukuran kelompok atau kawanan (jumlah partikel) adalah N . Untuk mengurangi jumlah evaluasi fungsi yang diperlukan untuk menemukan solusi, sebaiknya ukuran N tidak terlalu besar, tetapi juga tidak terlalu kecil, agar ada banyak kemungkinan posisi menuju solusi terbaik atau optimal. Jika terlalu kecil, sedikit kemungkinan menemukan posisi partikel yang baik. Terlalu besar juga akan membuat perhitungan jadi panjang. Biasanya digunakan ukuran kawanan adalah 20 sampai 30 partikel.
2. Bangkitkan populasi awal X dengan rentang $X(B)$ dan $X(A)$ secara random sehingga didapat X_1, X_2, \dots, X_N . Setelah itu, untuk mudahnya, partikel j dan kecepatannya pada iterasi i dinotasikan sebagai $X(i)j$ dan $V(i)j$ Sehingga partikel-partikel awal ini akan menjadi $X_1(0), X_2(0), \dots, X_N(0)$. Vektor $X_j(0)$, ($j = 1, 2, \dots, N$) disebut partikel atau vektor koordinat dari partikel. (seperti kromosom dalam

algoritma genetika). Evaluasi nilai fungsi tujuan untuk setiap partikel dan nyatakan dengan

$$f[X1(0)], f[X2(0)], ..., f[XN(0)] \quad (2.18)$$

3. Hitung kecepatan dari semua partikel. Semua partikel bergerak menuju titik optimal dengan suatu kecepatan. Awalnya semua kecepatan dari partikel diasumsikan sama dengan nol. Set iterasi $i = 1$.
4. Pada iterasi ke- i , temukan 2 parameter penting untuk setiap partikel j yaitu:

- (a) Nilai terbaik sejauh ini dari $Xj(i)$ (koordinat partikel j pada iterasi i) dan nyatakan sebagai $Pbest, j$, dengan nilai fungsi obyektif paling rendah (kasus minimasi), $f[Xj(i)]$, yang ditemui sebuah partikel j pada semua iterasi sebelumnya. Nilai terbaik untuk semua partikel $Xj(i)$ yang ditemukan sampai iterasi ke- i , $Gbest$, dengan nilai fungsi tujuan paling kecil/minimum diantara semua partikel untuk semua iterasi sebelumnya, $f[Xj(i)]$.
- (b) Hitung kecepatan partikel j pada iterasi ke i dengan rumus sebagai berikut:

$$Vj(i) = Vj(i-1) + c1r1[Pbest,j - xj(i-1)] + c2r2[Gbest - xj(i-1)], j = 1, 2, ..., N \quad (2.19)$$

dimana $c1$ dan $c2$ masing-masing adalah *learning rates* untuk kemampuan individu (cognitive) dan pengaruh sosial (group), dan $r1$ dan $r2$ bilangan random yang berdistribusi uniform dalam interval 0 dan 1. Jadi parameters $c1$ dan $c2$ menunjukkan bobot dari memory (position) sebuah partikel terhadap memory (posisi) dari kelompok (swarm). Nilai dari $c1$ dan $c2$ biasanya adalah 2 sehingga perkalian $c1r1$ dan $c2r2$ memastikan bahwa partikel-partikel akan mendekati target sekitar setengah selisihnya.

- (c) Hitung posisi atau koordinat partikel j pada iterasi ke- i dengan cara

$$Xj(i) = Xj(i-1) + Vj(i); j = 1, 2, ..., N \quad (2.20)$$

Evaluasi nilai fungsi tujuan untuk setiap partikel dan nyatakan sebagai

$$f[X1(i)], f[X2(i)], ..., f[XN(i)] \quad (2.21)$$

5. Cek apakah solusi yang sekarang sudah konvergen. Jika posisi semua partikel menuju ke satu nilai yang sama, maka ini disebut konvergen. Jika belum konvergen maka langkah 4 diulang dengan memperbarui iterasi $i = i + 1$, dengan cara menghitung nilai baru dari $Pbest, j$ dan $Gbest$. Proses iterasi ini dilanjutkan sampai semua partikel menuju ke satu titik solusi yang sama. Biasanya akan ditentukan dengan kriteria penghentian (stopping criteria), misalnya jumlah selisih solusi sekarang dengan solusi sebelumnya sudah sangat kecil.

2.5 Random Drift Particle Swarm Optimization

Hasil analisis membuktikan bahwa nilai konvergen dari algoritma *Particle Swarm Optimization* dapat dicapai apabila setiap iterasi konvergen terhadap local focusnya, $p_{i,n} = (p_{i,n}^1, p_{i,n}^2, \dots, p_{i,n}^N)$ didefinisikan oleh koordinat:

$$p_{i,n}^j = \varphi_{i,n}^j Y_{i,n}^j + (1 - \varphi_{i,n}^j) G_n^j \quad \varphi_{i,n}^j \sim U(0,1) \quad (2.22)$$

dimana $\varphi_{i,n}^j = c_1 r_{i,n}^j / (c_1 r_{i,n}^j + c_2 R_{i,n}^j)$ dengan nilai acak (random) untuk $r_{i,n}^j$ dan juga $R_{i,n}^j$. Karena koefisien percepatan c_1 dan c_2 pada umumnya ditetapkan pada nilai yang sama, $\varphi_{i,n}^j$ adalah serangkaian dari *uniformly distributed random numbers* pada nilai (0,1).

Perpindahan arah partikel menuju $p_{i,n}$ adalah sama dengan pergerakan drift dari suatu elektron pada konduktor besi yang ditempatkan pada medan magnet eksternal. Akan tetapi, berdasarkan model elektron bebas, selain pergerakan drift disebabkan oleh medan elektrik, elektron juga berada di pergerakan termal, yang muncul menjadi pergerakan yang acak. Efek keseluruhan dari perpindahan elektron adalah elektron tersebut bergerak miring ke lokasi dengan energi potensial minimum. Hal ini terlihat jelas bahwa perpindahan elektron adalah analog dengan proses minimum mendapatkan solusi dari masalahnya, jika posisi elektron dianggap sebagai solusi kandidat dan fungsi energy potensial sebagai fungsi objektif.

Terinspirasi dari fakta yang ada di atas, dapat diasumsikan bahwa partikel yang terdapat di dalam *Random Drift Particle Swarm Optimization* (RDPSO) berperilaku seperti elektron yang berpindah ke

konduktor besi yang ditempatkan pada medan elektrik eksternal. Oleh karena itu, perpindahan dari partikel adalah superposisi dari pergerakan termal pergerakan drift menuju $p_{i,n}$. Berhubungan dengan hal tersebut, kecepatan partikel dapat direpresentasikan dengan $V_{i,n}^j = VR_{i,n}^j + VD_{i,n}^j$, dimana $VR_{i,n}^j$ dan $VD_{i,n}^j$ merepresentasikan kecepatan dari pergerakan termal dan pergerakan drift menuju $p_{i,n}$. Pada Tugas Akhir ini, didapatkan rumus untuk $VR_{i,n}^j$ dan $VD_{i,n}^j$ sebagai berikut:

$$VR_{i,n+1}^j = \sigma_{i,n}^j \lambda_{i,n+1}^j \quad (2.23)$$

$$VD_{i,n+1}^j = \beta(p_{i,n}^j - X_{i,n}^j) \quad (2.24)$$

Pada persamaan (3.8), $\lambda_{i,n+1}^j$ adalah nilai acak dengan standar distribusi normal dan $\sigma_{i,n}^j$ adalah standar deviasi pada distribusi Gaussian, dimana nilainya dideterminasi secara adaptif dengan:

$$\sigma_{i,n}^j = \alpha |C_n^j - X_{i,n}^j| \quad (2.25)$$

dimana $C_n^j = \left(\frac{1}{M}\right) \sum_{i=1}^M Y_{i,n}^j$, ($1 \leq j \leq N$) dan disebut sebagai *mean best position*. Pada persamaan (2.24), $p_{i,n}^j$ didefinisikan oleh (2.22). Pada persamaan (2.23) dan (2.24), α dan β adalah dua angka real positif menggunakan koefisien *thermal* dan koefisien *drift*. Hal ditunjukkan dengan kecepatan pada pergerakan termal $VR_{i,n}^j$ mengikuti hukum kecepatan distribusi Maxwell. Hal ini juga dapat dilihat pada persamaan (3.9) bahwa efek dari kecepatan drift $VD_{i,n}^j$ memimpin partikel untuk membuat perpindahan menuju $p_{i,n}^j$ pada setiap iterasi. Dengan spesifikasi-spesifikasi yang sudah ada, didapatkan persamaan untuk partikel pada *Random Drift Particle Swarm Optimization*, yaitu:

$$V_{i,n}^j = \alpha |C_n^j - X_{i,n}^j| \lambda_{i,n+1}^j + \beta(p_{i,n}^j - X_{i,n}^j) \quad (2.26)$$

$$X_{i,n+1}^j = X_{i,n}^j + V_{i,n+1}^j \quad (2.27)$$

Prosedur dalam percobaan algoritma *Random Drift Particle Swarm Optimization* adalah sebagai berikut:

Step 0 : menginisiasi posisi saat ini pada setiap partikel dan meng-setting setiap *personal best position* untuk menjadi posisi saat ini;

Step 1 : setting $n = 0$;

- Step 2 : sementara kondisi yang diinginkan belum didapatkan, ikuti langkah-langkah selanjutnya;
- Step 3 : setting $n = n + 1$ dan hitung *mean best position* C_n ;
- Step 4 : dari $i = 1$ mengeksekusi langkah-langkah berikut;
- Step 5 : mengevaluasi nilai dari $f(X_{i,n})$, dan meng-update $Y_{i,n}$ dan G_n
- Step 6 : meng-update setiap komponen pada kecepatan dan posisi saat ini pada partikel i berdasarkan persamaan (3.11) dan (3.12);
- Step 7 : setting $i = i + 1$ dan kembali pada step 5 sampai $i = M$;
- Step 8 : kembali ke step 2.

2.6. Jenis-Jenis Gangguan Osilasi

Didalam permasalahan kestabilan dinamik, terdapat beberapa jenis gangguan osilasi yang menghasilkan frekuensi frekuensi tertentu, frekuensi tersebut memengaruhi pembentukan eigenvalue pada sebuah state space sistem yang diamati. Pada akhirnya, osilasi-osilasi ini akan memengaruhi respon kestabilan dinamik pasca dipasangkan PSS ataupun kontroler lainnya [9].

2.6.1. Mode Osilasi *Intraplant*

Mesin di lokasi pembangkit listrik yang sama akan berosilasi terhadap satu sama lain pada frekuensi 2.0-3.0 Hz, hal ini tergantung pada rating unit dan reaktansi yang menghubungkan mereka. Osilasi ini disebut sebagai *intra*

plant karena osilasi menampakkan diri di dalam kompleks pembangkit. Sisa dari sistem tidak terpengaruh.

2.6.2. Mode Osilasi *Local Area*

Dalam mode lokal, satu generator terayun oleh sisa sistem keseluruhan pada 1,0-2,0 Hz. Dampak dari osilasi terlokalisir ke generator dan saluran yang menghubungkannya ke grid. Sisa dari sistem biasanya dimodelkan sebagai sumber tegangan konstan yang frekuensi diasumsikan tetap konstan. Hal ini dikenal sebagai model Single Machine Infinite Bus (SMIB). Redaman dan frekuensi bervariasi berdasarkan output mesin dan impedansi antara terminal mesin dan tegangan infinite bus. Osilasi dapat dihapus dengan PSS masukan tunggal atau ganda yang menyediakan modulasi tegangan referensi dari regulator tegangan otomatis (AVR) dengan fase yang tepat dan gain compensation circuit

2.6.3. Mode Osilasi *Interarea*

Fenomena ini diamati pada sebagian besar jaringan. Mode osilasi *interarea* melibatkan dua kelompok koheren dari generator-generator yang berayun terhadap satu sama lain pada frekuensi 1 Hz atau kurang. Frekuensi osilasi sekitar 0,3 Hz. Fenomena yang kompleks ini melibatkan banyak bagian dari sistem dengan perilaku dinamis dan sifat non-linear. Karakteristik redaman dari mode *interarea* ditentukan oleh kekuatan *tie-line*, sifat beban, daya yang mengalir ke sistem interkoneksi dan interaksi beban dengan dinamika generator dan kontrol terkait.

2.6.4. Mode Osilasi Kontrol

Kemunculan osilasi ini disebabkan oleh hubungan antara generator, sistem *exciter*, *governor*, konverter HVDC dan kontrol SVC yang *ter-setting* dengan tidak baik. Beban dan sistem eksitasi dapat saling berinteraksi melalui mode kontrol. Kontrol tap trafo juga dapat berinteraksi dengan cara yang kompleks dengan beban non-linear sehingga menimbulkan osilasi tegangan.

2.6.5. Mode Osilasi Torsional

Mode ini berhubungan dengan sistem poros turbin generator dalam rentang frekuensi 10-46 Hz. Biasanya mode ini dapat tereksitasi ketika generator turbin multi-tahap terhubung ke sistem grid melalui serangkaian saluran yang terkompensasi secara seri. Mode torsi mekanik dari sistem poros berinteraksi dengan kapasitor seri pada frekuensi dasar jaringan listrik. Resonansi poros muncul ketika jaringan frekuensi sistem sama dengan frekuensi sinkron dikurangi frekuensi torsi.

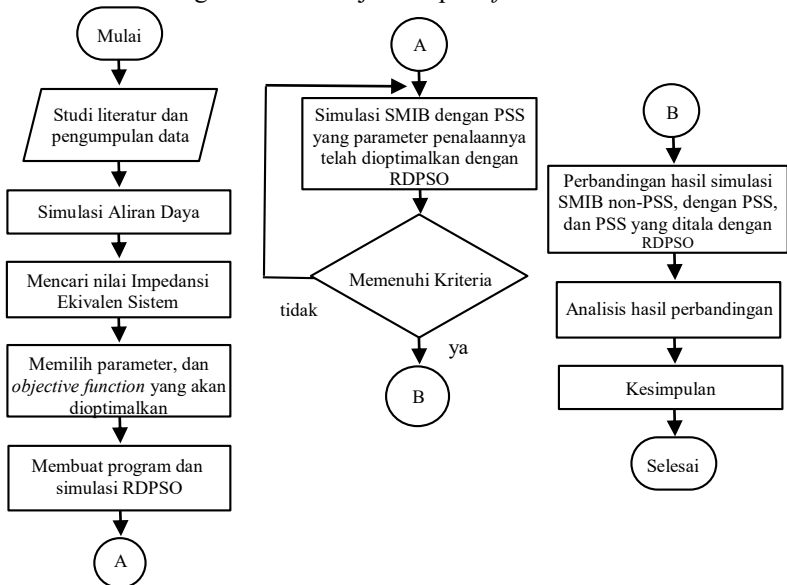
BAB III

PENALAAN OPTIMAL PSS SMIB UNIT II PLTU PACITAN PADA SISTEM 150 KV JAWA TIMUR

Pada Tugas Akhir ini, *Power System Stabilizer* (PSS) digunakan pada sistem *Single Machine Infinite Bus* (SMIB). Untuk mendapatkan nilai impedansi ekivalen didapatkan dengan metode *losses concept*, data pembangkitan dan pembebanan dijalankan pada *script* MATLAB untuk mendapatkan aliran daya pada sistem. Sedangkan untuk mendapatkan koordinasi yang optimal, parameter PSS dioptimalkan dengan menggunakan algoritma *Random Drift Particle Swarm Optimization* (RDPSO). Simulasi ini dilakukan menggunakan *software* bantu MATLAB.

3.1 Alur Penyelesaian Tugas Akhir

Prosedur Tugas Akhir ditunjukkan pada *flowchart* dibawah ini :



Gambar 3.1 *Flowchart* Pengerjaan Tugas Akhir

Pengerjaan Tugas Akhir ini dimulai dengan studi literatur dan pengumpulan data dari referensi yang ada. Referensi yang dimaksud berhubungan dengan pokok bahasan SMIB, kestabilan dinamik, PSS dan algoritma RDPSO. Langkah selanjutnya adalah memodelkan sistem tenaga listrik. Sistem tenaga listrik yang dimodelkan adalah pembangkit Pacitan. Data aliran daya didapatkan dari rekapitulasi data utama pada MATLAB. Dari hasil *running*, langkah selanjutnya adalah melakukan penghitungan nilai *Req* dan *Xeq* dengan menggunakan *losses concept*.

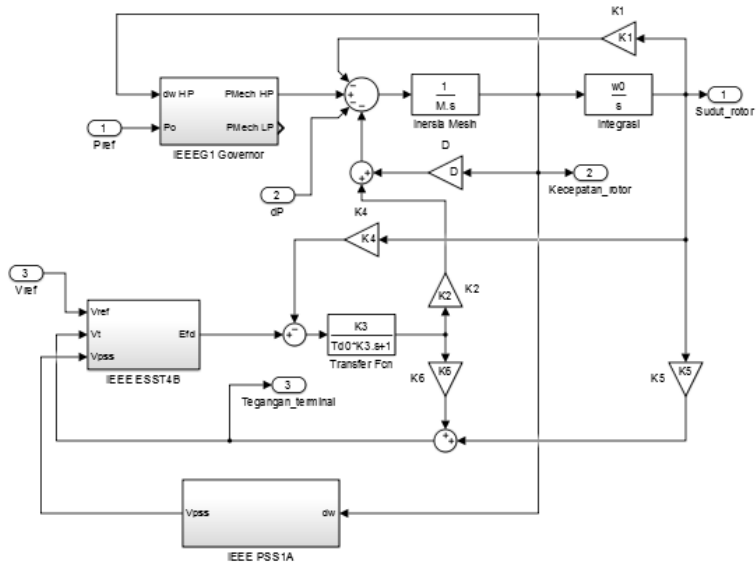
Terdapat tiga macam permodelan pada Tugas Akhir ini : pemodelan sistem SMIB tanpa PSS, pemodelan sistem SMIB dengan menggunakan PSS yang tidak dioptimalkan, dan permodelan sistem SMIB yang dioptimalkan dengan RDPSO untuk mendapatkan respon *overshoot* dan *settling time* yang baik.

Hasil simulasi dari ketiga permodelan diatas akan di analisis. Respon sistem yang diamati adalah respon perubahan frekuensi, dan respon perubahan sudut rotor akibat perubahan beban 0,01 p.u, dan 0,05 p.u. Dari hasil respon sistem yang didapat, akan dianalisis seberapa besar perbaikan redaman yang dihasilkan oleh penggunaan PSS yang ditala dengan metode metode RDPSO dalam memperbaiki nilai *overshoot* dan *settling time*. Permodelan PSS dan pengerjaannya akan dijelaskan pada sub bab dibawah ini.

3.2 Permodelan Single Machine Infinite Bus Unit II PLTU Pacitan

3.2.1 Permodelan Mesin, Sistem Eksitasi, dan PSS

Pada Tugas Akhir ini, permodelan *Heffron Phillips* dipilih untuk memodelkan mesin dan sistem eksitasi Unit II PLTU Pacitan. Gambar 3.2 menunjukkan permodelan *Heffron Phillips* dengan *Simulink*. Permodelan ini dilengkapi dengan PSS. Data yang didapatkan dari Pusat Pengaturan Beban Jawa-Bali menyatakan bahwa governor dan turbin dimodelkan dengan model PSS/E IEEE1, dan *exciter* dengan model PSS/E ESST4B. Permodelan yang dilakukan dengan menggunakan *Simulink* untuk model *governor*, turbin, dan *exciter* terlampir pada bagian lampiran.



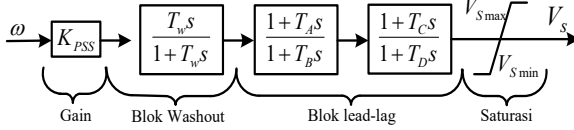
Gambar 3.2 Model Generator dan Sistem Eksitasi Unit II PLTU Pacitan dengan PSS

Gambar 3.2 adalah gambar model mesin dan sistem eksitasi SMIB Unit II Pacitan dengan tambahan PSS. Simulasi dilakukan dengan memasang PSS pada SMIB. Dalam simulasi diamati respon perubahan frekuensi dan respon perubahan sudut rotor saat terjadi gangguan perubahan beban pada pembangkit sebesar 0,01 p.u, dan 0,05 p.u.

dengan :

- D = Koefisien Redaman
- w_0 = kecepatan sinkron mesin
- T_d0 = konstanta waktu pada model belitan medan
- K_3 = konstanta penguat pada model belitan medan
- M = representasi dari konstanta Inersia pada mesin (2H)
- dP = model perubahan beban (disambungkan pada unit step)
- v_{ref} = tegangan referensi yang bernilai = 0

Model PSS ditunjukkan pada gambar 3.3. Penjelasan model PSS berikut dapat dilihat pada Bab II pada bagian penjelasan *Power System Stabilizer*



Gambar 3.3 Model PSS pada Unit II PLTU Pacitan

dengan :

- K_{pss} = Konstanta Gain
- T_w = Konstanta waktu pada bagian *wash out*
- T_1 = konstanta waktu pada 1 blok diagram *lead-lag*
- T_2 = konstanta waktu 2 pada blok diagram *lead-lag*
- T_3 = konstanta waktu 3 pada blok diagram *lead-lag*
- T_4 = konstanta waktu 4 pada blok diagram *lead-lag*

3.2.2 Konstanta K

Konstanta K adalah bagian yang tak terpisahkan dengan model generator dan sistem eksitasi. Masing-masing K memiliki peran penting dan hubungan yang erat dengan beberapa kejadian yang disebabkan oleh peralatan kontrol tertentu pada generator.

3.2.2.1 Konstanta K1 dan K2

Konstanta K1 dan K2 berhubungan dengan persamaan variabel torsi elektrik. Persamaan torsi elektrik ΔT_e dalam bentuk linier mesin sinkron ditulis sebagai berikut [2]:

$$\begin{aligned} \Delta T_e = & K_I V_\infty E_{q0} [R_e \sin(\delta_0 - \alpha) + (x'_d + X_e) \cos(\delta_0 - \alpha)] \Delta \delta \\ & + K_I V_\infty I_{q0} (x_q - x'_d) [(x_q - X_e) \sin(\delta_0 - \alpha) - R_e \cos(\delta_0 - \alpha)] \Delta \delta \\ & + K_I I_{q0} [R_e^2 + (x_q + X_e)^2] \Delta E'_q + K_I E_{q0} R_e \Delta E'_q \end{aligned} \quad (3.1)$$

Dari persamaan 3.1 maka ΔT_e terdiri dari komponen $\Delta \delta$ dan $\Delta E'_q$ sehingga dapat didekati dengan persamaan berikut:

$$\Delta T_e = K_1 \Delta \delta + K_2 \Delta E'_q \quad (3.2)$$

K_1 adalah perubahan torsi elektrik untuk perubahan kecil sudut rotor pada fluks sumbu *direct* konstan dan K_2 adalah perubahan torsi elektrik untuk perubahan kecil fluks sumbu *direct* pada sudut rotor konstan, sehingga

dari persamaan 3.2, konstanta K_1 dan K_2 dapat didefinisikan sebagai berikut:

$$K_1 = \left. \frac{\Delta T e}{\Delta \delta} \right|_{E'q=E'q_0} \quad (3.3)$$

Dari persamaan 3.2 juga dapat diperoleh,

$$K_1 = K_I V_{\infty} E_{qa0} [R_e \sin(\delta_0 - \alpha) + (x'_d + X_e) \cos(\delta_0 - \alpha)] \\ + K_I V_{\infty} I_{q0} (x_q - x'_d) [(x_q - X_e) \sin(\delta_0 - \alpha) - R_e \cos(\delta_0 - \alpha)] \quad (3.4)$$

dengan

$$K_I = \frac{1}{R_e^2 + (x_q + X_e)(x'_d + X_e)} \quad (3.5)$$

Sedangkan untuk K_2 dapat didefinisikan sebagai berikut:

$$K_2 = \left. \frac{\Delta T e}{\Delta E'q} \right|_{\delta=\delta_0} \quad (3.6)$$

Dari persamaan 3.2 juga dapat diperoleh

$$K_2 = K_I I_{q0} [R_e^2 + (x_q + X_e)^2] + K_I E_{qa0} R_e \quad (3.7)$$

3.2.2.2 Konstanta K3 dan K4

Konstanta K_3 dan K_4 berhubungan persamaan variabel tegangan $\Delta E'q$. Dengan mengambil hubungan bahwa

$$E_{FD} = (1 + \tau'_{d0})E'_q - (x_d - x'_d)I_d \quad (3.8)$$

dan arus oleh Anderson dan Fouad dirumuskan sebagai

$$\begin{bmatrix} \Delta I_d \\ \Delta I_q \end{bmatrix} = K_I \begin{bmatrix} -(x_q + X_e) & R_e \cos(\delta_0 - \alpha) - (x_q - X_e) \sin(\delta_0 - \alpha) \\ R_e & (x'_d - R_e) \cos(\delta_0 - \alpha) + R_e \sin(\delta_0 - \alpha) \end{bmatrix} \begin{bmatrix} \Delta E'_q \\ V_{\infty} \Delta \delta \end{bmatrix} \quad (3.9)$$

Maka diperoleh

$$\Delta E_{FD} = \left(\frac{1}{K_3 + \tau'_{d0} s} \right) \Delta E'_q + K_4 \Delta \delta \quad (3.10)$$

Dengan nilai

$$K_3 = \frac{1}{1 + K_I (x_d - x'_d)(x_q + X_e)} \quad (3.11)$$

Dan nilai

$$K_4 = V_{\infty} K_I (x_d - x'_d) [(x_q + X_e) \sin(\delta_0 - \alpha) - R_e \cos(\delta_0 - \alpha)] \quad (3.12)$$

Dari persamaan 3.10 dapat diketahui bahwa K_3 adalah faktor impedansi yang dibawa ke perhitungan efek beban dari impedansi eksternal, dan K_4 berhubungan dengan efek demagnetisasi dari perubahan sudut rotor, sehingga keduanya dapat ditulis sebagai

$$K_4 = \frac{1}{K_3} \frac{E' q}{\Delta \delta} \Big|_{E_{FD} = \text{konstan}} \quad (3.13)$$

3.2.2.3 Konstanta K5 dan K6

Untuk menghitung konstanta K_5 dan K_6 dibutuhkan hubungan dengan variabel tegangan terminal ΔV_t yang dalam bentuk persamaan linier diberikan sebagai berikut:

$$\begin{aligned} \Delta V_t = & K_I V_{\infty} x'_d \frac{V_{q0}}{V_{t0}} [R_e \cos(\delta_0 - \alpha) - (x_q + X_e) \sin(\delta_0 - \alpha)] \Delta \delta \\ & - K_I V_{\infty} x_q \frac{V_{d0}}{V_{t0}} [(x'_d + X_e) \cos(\delta_0 - \alpha) + R_e \sin(\delta_0 - \alpha)] \Delta \delta \\ & + \frac{V_{q0}}{V_{t0}} [1 - K_I x'_d (x_q + X_e)] \Delta E'_q - \frac{V_{d0}}{V_{t0}} K_I x_q R_e \Delta E'_q \end{aligned} \quad (3.14)$$

Persamaan 3.14 dapat disejajarkan dengan

$$\Delta V_t = K_5 \Delta \delta + K_6 \Delta E'_q \quad (3.15)$$

Dari persamaan 3.20 3.15, K_5 adalah perubahan tegangan terminal ΔV_t untuk perubahan kecil dari sudut rotor pada fluks sumbu *direct* konstan, yang dapat ditulis sebagai

$$\begin{aligned} K_5 = & K_I V_{\infty} x'_d \frac{V_{q0}}{V_{t0}} [R_e \cos(\delta_0 - \alpha) - (x_q + X_e) \sin(\delta_0 - \alpha)] \\ & + K_I V_{\infty} x_q \frac{V_{d0}}{V_{t0}} [(x'_d + X_e) \cos(\delta_0 - \alpha) + R_e \sin(\delta_0 - \alpha)] \end{aligned} \quad (3.16)$$

atau dalam bentuk ringkas dapat ditulis sebagai

$$K_5 = \frac{\Delta V_t}{\Delta \delta} \Big|_{E'_q = E'_{q0}} \quad (3.17)$$

Sedangkan konstanta K_6 adalah perubahan tegangan terminal V_t untuk perubahan kecil dari fluks sumbu *direct* pada sudut rotor konstan. Dari persamaan 3.14 dapat ditulis

$$K_6 = \frac{V_{q0}}{V_{t0}} [1 - K_I x'_d (x_q + X_e)] - \frac{V_{d0}}{V_{t0}} K_I x_q R_e \quad (3.18)$$

atau dalam bentuk ringkas ditulis sebagai

$$K_6 = \left. \frac{\Delta V_e}{\Delta E'_{tq}} \right|_{\delta=\delta_0} \quad (3.19)$$

Dengan menggunakan data kondisi inisial tegangan dan arus yang sudah tersedia, dan juga data dari mesin dan jaring, maka konstanta K_1 sampai dengan K_6 dengan mudah dapat dicari.

Keterangan persamaan semua di atas:

- V_{∞} :Tegangan bus tak berhingga (karena sebagai tegangan referensi dianggap 1,0 p.u)
- V_{q0} :Tegangan output generator sumbu *quadrature* saat *steady state* (diperoleh dari aliran daya)
- V_{d0} :Tegangan output generator sumbu *direct* saat *steady state* (diperoleh dari aliran daya)
- V_{t0} :Tegangan output total generator saat *steady state* (diperoleh dari aliran daya)
- E_{qa0} :Tegangan internal generator saat *steady state* (diperoleh dari aliran daya)
- R_e :Jumlah resistansi saluran dan trafo antara generator dan bus tak berhingga
- α :Sudut antara tegangan dan arus generator
- δ_0 :Sudut daya rotor generator saat *steady state* (diperoleh dari aliran daya)
- x_d :Reaktansi stator generator sumbu *direct*
- x'_d :reaktansi transient generator sumbu *direct*
- X_e :Jumlah reaktansi saluran dan trafo antara generator dan bus tak berhingga
- I_{q0} :Arus stator generator saat *steady state* (diperoleh dari aliran daya)
- x_q :Reaktansi stator generator sumbu *quadrature*
- x'_q :Reaktansi transient stator generator sumbu *quadrature*

3.3 Data Pembangkit, dan Saluran SMIB Unit II Pacitan pada Sistem Kelistrikan 150 kV Jawa Timur

Berikut adalah data mesin Unit II PLTU Pacitan dan saluran 150 kV Jawa Timur. Data saluran dihitung dengan *Losses Concept*.

Tabel 3.1 Data Pembangkitan Unit II PLTU Pacitan

| Pembangkit (MVA Dasar = 1000 MVA) | Besar Pembangkitan | |
|---|--------------------|---------|
| | MW | MVAR |
| Unit II Pacitan | 150,037 | -17,403 |

Tabel 3.2 Parameter Dinamik Unit II PLTU Pacitan

| Pembangkit | Xq (p.u) | Xq' (p.u) | Xd (p.u) | Xd' (p.u) | $rstr$ |
|------------|------------|-------------|------------|-------------|-------------|
| Unit II | 0,083699 | 0,014348 | 0,08848 | 0,01294 | 0,000176965 |

Tabel 3.3 Data Saluran Sistem 150 kV Jawa Timur

| Saluran (R_{eq} dan X_{eq}) | $r + jx$ (p.u) |
|-----------------------------------|---------------------------|
| | 0,028369801+ j0,129260477 |

Tabel 3.4 Data Mesin

| Parameter Mesin | | | |
|-----------------|-------|-----------------|----------------|
| Variabel | Nilai | Variabel | Nilai |
| H | 4,114 | $R_{eq}+X_{eq}$ | 0,0284+i0,1293 |
| D | 7 | P | 0,150037 |
| $Td0$ | 0 | Q | -0,017403 |
| $\omega0$ | 314 | V_t | 1,00 |

Tabel 3.5 Data AVR

| Data AVR | | | |
|----------|-------|------------|-------|
| Variabel | Nilai | Variabel | Nilai |
| Tr | 0,01 | VR_{Max} | 9,1 |
| Kpr | 33 | VR_{Min} | -7,1 |
| Kir | 3 | V_{MMax} | 9,1 |
| Ta | 0,05 | V_{MMin} | -7,1 |

Tabel 3.6 Data Governor dan Turbin

| Data Governor dan Turbin | | | |
|--------------------------|-------|----------|-------|
| Variabel | Nilai | Variabel | Nilai |
| K | 10 | $K1$ | 0,1 |
| $T1$ | 0,5 | $K3$ | 0,1 |
| $T2$ | 0,1 | $K5$ | 0,3 |
| $T3$ | 1 | $K7$ | 0,5 |
| $T4$ | 1,2 | Uo | 1,2 |
| $T5$ | 1,2 | Uc | 0 |
| $T6$ | 1,2 | Pmax | 0,315 |
| $T7$ | 1,2 | Pmin | 0 |

Jenis dari Governor dan turbin dalam permodelan adalah tipe IEEE G1 *Turbine Governor Model*, dengan Exciter berjenis PSS/E Model ESST4B. Jenis PSS yang dipilih adalah jenis PSS1A. Gambar dari Governor, dan Exciter terlampir di bagian lampiran.

Sistem yang digunakan dalam simulasi Tugas Akhir ini adalah model mesin Unit II PLTU Pacitan yang dipasang dengan PSS. PSS berfungsi untuk menghasilkan sinyal kontrol untuk diumpankan pada sistem eksitasi. Untuk meredam osilasi frekuensi dan sudut rotor ke kondisi yang stabil maka harus ditentukan parameter dari PSS tersebut. Data parameter PSS yang belum dioptimalkan ditunjukkan pada Tabel 3.7 [7]

Tabel 3.7 Data PSS yang belum dioptimalkan

| Parameter | Nilai penalaan |
|---------------|----------------|
| K_{pss} | 10,62 |
| $sTw/(1+sTw)$ | 1,00 |
| $T1$ | 0,911 |
| $T2$ | 0,034 |
| $T3$ | 10,071 |
| $T4$ | 0,2 |

3.4 State Space dan Eigenvalue

Pada Tugas Akhir ini, *state space* digunakan untuk mencari nilai *eigenvalue* pada sistem yang dimodelkan. Pemodelan matriks keadaan (*state space*) berfungsi untuk mempermudah dalam penyelesaian orde tinggi dari rumus matematis sistem. Penggunaan *state space* juga dapat digunakan untuk melihat respon dari sistem. *State space* SMIB didapat dengan menggunakan fungsi linierisasi *linmod* pada MATLAB. *State Space* akan diperbaharui di tiap iterasi pada proses optimisasi seiring dengan *update* kecepatan partikel RDPSO yang terjadi. *State Space* pada iterasi pertama didapat dengan menginisiasi parameter PSS secara acak. Matriks yang digunakan untuk melakukan analisis *eigenvalue* adalah matriks A. Melalui matriks sistem A, nilai *eigenvalue* sistem dapat dipantau :

$$\det(sI-A)=0 \quad (3.20)$$

Nilai I adalah matriks identitas, dan nilai s adalah *eigenvalue* dari sistem Matriks A berukuran $n \times n$, sehingga banyaknya *eigenvalue* yang diperoleh sebanyak n , $\lambda = \lambda_1, \lambda_2, \lambda_3, \dots, \lambda_n$.

$$\lambda_i = \sigma_i + j\omega_i \quad (3.21)$$

Frekuensi osilasi dalam Hz adalah

$$\text{Frekuensi osilasi} = \frac{\omega_i}{2\pi} \quad (3.22)$$

Dengan,

$$\begin{aligned} \lambda_i &= \text{Eigenvalue ke-}i \\ \sigma_i &= \text{Komponen riil dari eigenvalue ke-}i \\ \omega_i &= \text{Komponen imajiner dari eigenvalue ke-}i \end{aligned}$$

Bagian riil *eigenvalue* merupakan komponen redaman, sedangkan bagian imajiner merupakan komponen osilasi. Nilai redaman dapat diketahui menggunakan nilai *damping ratio*. Sedangkan redaman sistem secara keseluruhan dapat diketahui dari nilai *Comprehensive Damping Index* (CDI). Pada Tugas Akhir ini, CDI akan menjadi fungsi objektif pada proses optimisasi penalaan PSS.

3.5 Penghitungan Nilai Impedansi Ekuivalen dengan *Losses Concept*

Pada Tugas Akhir ini, sistem transmisi 150 kV 50 Hz Jawa Timur digunakan dalam studi kasus, dengan PLTU Pacitan sebagai mesin yang diamati. Sistem 150 kV merupakan sistem interkoneksi yang cukup rumit, sehingga dengan metode reduksi konvensional tentu membutuhkan perhitungan yang rumit, maka dari itu, reduksi jaringan menggunakan *losses concept* merupakan sebuah solusi yang cukup baik dan simpel. Berikut adalah langkah-langkah reduksi jaringan menggunakan *losses concept* [10]:

1. Lakukan aliran daya sistem interkoneksi dengan banyak mesin (multimesin). Perhatikan nilai pembebanan pada mesin yang akan diamati. Dari hasil studi aliran daya dengan menggunakan MATLAB, nilai dari daya aktif Pembangkitan Pacitan, daya aktif pembebanan Pacitan, daya aktif total pembangkitan, daya reaktif total pembangkitan, rugi rugi total pada sistem adalah sebagai berikut:

Tabel 3.8 Data Aliran Daya Unit II PLTU Pacitan

| Unit II Pacitan | Satuan | |
|-----------------|---------|----------|
| | MW | MVAR |
| Pembangkitan | 150 | 30 |
| Pembebanan | 15,9954 | 3,491961 |

Tabel 3.9 Data Aliran Daya Sistem 150 kV Jawa Timur

| Sistem 150 kV Jawa Timur | Satuan | |
|--------------------------|----------|----------|
| | MW | MVAR |
| Pembangkitan Total | 5197,824 | 3036,255 |
| Pembebanan Total | 5125,427 | 2030,404 |
| Rugi-Rugi Total | 72,397 | 325,851 |

2. Siapkan nilai data dengan mengatur pembebanan mesin yang diamati sehingga tidak melebihi kapasitas mula-mula dari hasil aliran daya sistem interkoneksi. Hal ini berguna untuk mendekati kondisi operasi mesin yang diamati. Lakukan aliran daya dengan semua mesin tidak menyala, kecuali mesin yang diamati. Mesin yang diamati diganti menjadi mode *slack bus*.

- Lakukan penyekalaan pada data pembebanan pada masing masing bus, sehingga nilai dari total pembebanan pada saat semua generator dimatikan (kecuali Pacitan) tidak melebihi titik operasi pembangkit Pacitan. Penyekalaan dilakukan dengan :

$$Pload\ baru = \frac{P_{gen}}{P_{total}} \times Ploadnormal \times 1.1 \quad (3.23)$$

$$Qload\ baru = \frac{Q_{gen}}{Q_{total}} \times Ploadnormal \times 1.4 \quad (3.24)$$

Dengan,

- Pload baru* : Daya aktif pada masing masing Bus setelah penyekalaan
- Qload baru* : Daya Reaktif pada masing masing Bus setelah penyekalaan
- Pgen* : Daya aktif Pembangkit Pacitan pada kondisi normal
- Qgen* : Daya Reaktif Pembangkit Pacitan pada kondisi normal
- Ploadnormal* : Daya aktif beban tiap bus pada kondisi normal
- Qloadnormal* : Daya Reaktif beban tiap bus pada kondisi normal

Nilai 1.1 dan 1.4 dipilih untuk merekayasa agar total nilai Pload dari masing-masing bus setelah dilakukan penyekalaan mendekati nilai titik operasi generator Pacitan sebesar 150,037 MW.

- Dapatkan daya aktif Pacitan, daya reaktif pacitan, rugi-rugi daya aktif sistem dan rugi-rugi daya reaktif sistem pada saat semua generator lain dimatikan, dan generator pacitan dikondisikan pada *slack bus*.

Tabel 3.10 Data Aliran Daya (Semua Mesin Dimatikan)

| Unit II Pacitan | Satuan | |
|------------------|---------|---------|
| | MW | MVAR |
| Pembangkitan | 150,037 | -17,403 |
| Pembebanan Total | 149,390 | -16,423 |
| Rugi-Rugi Total | 0,622 | 0,29 |

Dapat dilihat dari tabel diatas, nilai dari simulasi pembebanan pada sistem Jawa Timur 150 kV telah mendekati titik operasi pembangkit pacitan. Nilai dari rugi-rugi total yang dihasilkan oleh MATLAB kini seakan-akan merupakan nilai rugi-rugi yang berasal dari pembangkit pacitan saja. Nilai ini kemudian digunakan untuk mencari Impedansi Ekvaleen sistem untuk SMIB Pacitan.

5. Hitung nilai arus dari mesin yang diamati menggunakan persamaan :

$$I = \frac{MVA \text{ Mesin (p.u)}}{\sqrt{3} \cdot V_{\text{mesin (p.u)}}} \quad (3.25)$$

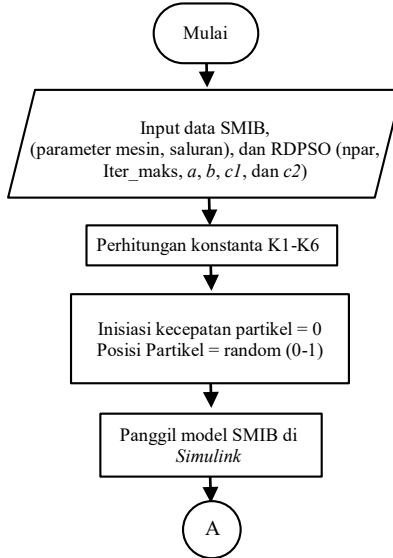
$$I = \frac{0,150037 + j0,017403}{\sqrt{3} \cdot 1,02 + 0,0i} = 0,0849 + j0,0099 \text{ p.u} \quad (3.26)$$

6. Hitung nilai Req dan Xeq sesuai dengan persamaan :

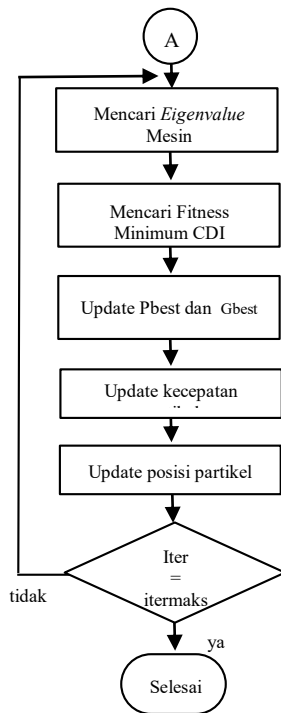
$$\begin{aligned} Z_{eq} = R_{eq} + jX_{eq} &= \frac{\text{Rugi} - \text{Rugi Total}}{I^2} = \frac{0.000622 + j0.0029}{|0.0849 + j0.0099|^2} \quad (3.27) \\ &= 0.0284 + j0.1293 \text{ p.u.} \end{aligned}$$

3.6 Penalaan Parameter PSS Menggunakan RDPSO

Berikut adalah *Flowchart* Algoritma RDPSO pada Penalaan PSS.



Gambar 3.4 Algoritma RDPSO pada Penalaan Optimal Bagian I



Gambar 3.5 Algoritma RDPSO pada Penalaan Optimal PSS Bagian II

Tabel 3.11 adalah parameter RDPSO yang dipilih dalam proses pencarian parameter teroptimal untuk menala PSS.

Tabel 3.11 Variabel Masukan pada RDPSO

| | | | |
|-----------------|-----|--------------|------|
| <i>Npar</i> | 100 | <i>Amaks</i> | 0,9 |
| <i>Dimensi</i> | 6 | <i>B</i> | 1,45 |
| <i>Iter max</i> | 100 | <i>c1</i> | 2,05 |
| <i>Amin</i> | 0,3 | <i>c2</i> | 2,05 |

Dengan

- N_{par} = jumlah elektron
- $dimensi$ = dimensi elektron (berupa 6 buah parameter PSS)
- $Iter_maks$ = iterasi maksimum pada proses optimisasi
- $amin$ = konstanta thermal minimum
- $amaks$ = konstanta thermal minimum
- b = konstanta drift pada VD (perpindahan yang dipengaruhi oleh medan listrik)
- $c1, c2$ = konstanta acak untuk memodelkan gerak acak elektron yang bernilai 0-1

Dalam rangka mencari parameter PSS yang optimal, PSO untuk menala parameter PSS (K_{pss} , T_w , $T1$, $T2$, $T3$, dan $T4$). Hasil dari penalaan ini kemudian dianalisis untuk mendapatkan respon yang baik. Fungsi objektif yang digunakan untuk menguji kestabilan sistem adalah *Comprehensive Damping Index* (CDI) yang dirumuskan pada Persamaan 3.29 [8].

$$\zeta_i = \frac{-\sigma_i}{\sqrt{\sigma_i^2 + \omega_i^2}} \quad (3.28)$$

$$CDI = \sum_{i=1}^n (1 - \zeta_i) \quad (3.29)$$

Dengan,

ζ_i = Damping ratio ke-i

n = jumlah *eigenvalue*

Sebelum mencari CDI, *eigenvalue* dari model generator dan sistem eksitasi harus didapat terlebih dahulu. *Eigenvalue* didapat dengan cara menggunakan fungsi “linmod” pada *m.file*. Nilai-nilai dari parameter penalaan (K_{pss} , T_w , $T1$, $T2$, $T3$, dan $T4$) dibatasi oleh *constraint* yang tertera pada tabel 3.12 [7].

Tabel 3.12 Interval Parameter Penalaan PSS

| Parameter | Nilai Maksimum | Nilai Minimum |
|-----------|----------------|---------------|
| K_{pss} | 50 | 0,01 |
| T_w | 5 | 4,5 |
| $T1$ | 1,5 | 0,06 |

Tabel 3.12 Interval Parameter Penalaan PSS

| | | |
|------|-----|------|
| $T2$ | 0,1 | 0,01 |
| $T3$ | 1,5 | 0,06 |
| $T4$ | 0,1 | 0,01 |
| $T2$ | 0,1 | 0,01 |

Program *m.file* pada MATLAB akan terus berjalan hingga mencapai iterasi maksimum. Nilai *update* dari *pbest* dan *gbest* yang didapat dari *objective function* minimum CDI [8] yang direkam setiap iterasinya. Nilai tersebut akan berubah seiring dengan perubahan kecepatan pada kecepatan thermal (VR) dan kecepatan drift (VD). Resultan dari perubahan kecepatan thermal dan kecepatan Drift adalah perubahan posisi elektron yang kian mendekati daerah dengan nilai potensial terendah (titik teroptimal). Detail dari konstanta dan persamaan yang terlibat pada pembaharuan kecepatan disetiap iterasinya dijelaskan pada Bab II bagian *Random Drift Particle Swarm Optimization*. Hasil yang dihasilkan dari simulasi yaitu berupa perbandingan sistem tanpa PSS, sistem dengan PSS, dan sistem dengan PSS yang ditala menggunakan RDPSO. Respon sistem yang diamati adalah respon perubahan frekuensi, dan respon perubahan sudut rotor. Dari hasil respon tersebut penggunaan metode optimisasi PSS dengan RDPSO berfungsi untuk memperbaiki nilai *overshoot* dan *settling time*.

BAB IV

SIMULASI DAN ANALISIS

Pada Tugas Akhir ini, *Power System Stabilizer* (PSS) digunakan pada sistem *Single Machine Infinite Bus* (SMIB). Untuk mendapatkan koordinasi yang optimal, parameter PSS dioptimalkan dengan menggunakan algoritma RDPSO. Simulasi ini dilakukan menggunakan *software* bantu MATLAB pada *m.file* dan *Simulink*. Hasil penalaan optimal PSS dengan RDPSO menghasilkan nilai $K_{pss} = 36,1995$, $T_w = 4,9346$, $T_1 = 0,4869$, $T_2 = 0,0130$, $T_3 = 0,0703$, dan $T_4 = 0,0122$.

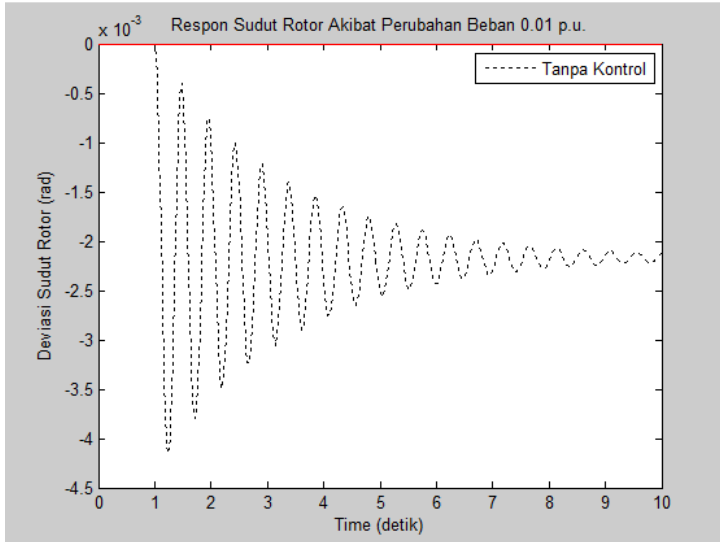
4.1 Analisis Eigenvalue

Matriks state space merupakan hasil linierisasi dari model generator dan eksitasi pada SMIB yang digunakan didalam Tugas Akhir ini. Komponen matriks A dari matriks *State Space* digunakan untuk menentukan nilai *eigenvalue* sistem. Nilai *eigenvalue* dari sistem SMIB tanpa PSS dapat dilihat pada tabel 4.1.

Tabel 4.1. Nilai *Eigenvalue* Dari Sistem SMIB Tanpa PSS

| No | <i>Eigenvalue</i> |
|----|-------------------|
| 1 | $-2,00 + 0,00i$ |
| 2 | $-100,86 + 0,00i$ |
| 3 | $-0,43 + 13,22i$ |
| 4 | $-0,43 - 13,22i$ |
| 5 | $-14,23 + 0,00i$ |
| 6 | $-4,99 + 0,00i$ |
| 7 | $-0,09 + 0,00i$ |
| 8 | $-0,00 + 0,00i$ |
| 9 | $-0,83 + 0,00i$ |
| 10 | $-0,83 + 0,00i$ |
| 11 | $-0,83 + 0,00i$ |
| 12 | $-0,83 + 0,00i$ |

Dari *eigenvalue* yang didapat, semua nilai konstanta redaman pada *eigenvalue* yang ada bernilai negatif. Dapat disimpulkan bahwa sistem sebelum ditambahkan dengan PSS berada pada kondisi stabil. Namun, pada gambar 4.1, dapat terlihat bahwa sinyal respon sudut rotor berosilasi pada saat diberi gangguan pelepasan beban 0,01 p.u. Hal ini diduga disebabkan oleh nilai redaman yang kecil pada sistem.



Gambar 4.1. Respon Sudut Rotor non-PSS pada Gangguan 0,01 p.u

Tabel 4.2 adalah tabel yang menunjukkan *eigenvalue* pada SMIB dengan PSS dan tidak dioptimalkan. Dari tabel dapat terlihat bahwa sistem juga dalam keadaan stabil.

Tabel 4.2. *Eigenvalue* SMIB-PSS
Tidak Dioptimalkan

| No | <i>Eigenvalue</i> |
|----|-------------------|
| 1 | $-2,00 + 0,00i$ |
| 2 | $-100,88 + 0,00i$ |
| 3 | $-40,54 + 0,00i$ |
| 4 | $-2,54 + 20,87i$ |

Tabel 4.2. *Eigenvalue* SMIB-PSS
Tidak Dioptimalkan

| | |
|----|------------------|
| 5 | $-2,54 - 20,87i$ |
| 6 | $-2,78 + 4,81i$ |
| 7 | $-2,78 - 4,81i$ |
| 8 | $-3,31 + 0,00i$ |
| 9 | $-0,09 + 0,00i$ |
| 10 | $0,00 + 0,00i$ |
| 11 | $-0,83 + 0,00i$ |
| 12 | $-0,83 + 0,00i$ |
| 13 | $-0,83 + 0,00i$ |
| 14 | $0,83 + 0,00i$ |

Tabel 4.3. *Eigenvalue* dari sistem SMIB-PSS-RDPSO

| No | <i>Eigenvalue</i> | <i>Frekuensi osilasi</i> |
|----|---------------------|--------------------------|
| 1 | $-2,00 + 0.00i$ | 0 |
| 2 | $-108,88 + 0.00i$ | 0 |
| 3 | $-99,28 + 0.00i$ | 0 |
| 4 | $-17,87 + 0.03.10i$ | 0,9882 |
| 5 | $-17,87 - 0.03.10i$ | -0,9882 |
| 6 | $-11,41 + 6.20i$ | 1,9745 |
| 7 | $-11,41 - 6.20i$ | -1,9745 |
| 8 | $-13,10 + 0.00i$ | 0 |
| 9 | $-0,20 + 0.00i$ | 0 |
| 10 | $-0,09 + 0.00i$ | 0 |
| 11 | $0,00 + 0.00i$ | 0 |
| 12 | $-0,83 + 0.00i$ | 0 |
| 13 | $-0,83 + 0.00i$ | 0 |
| 14 | $-0,83 + 0.00i$ | 0 |
| 15 | $-0,83 + 0.00i$ | 0 |

Tabel 4.3 adalah tabel *eigenvalue* dari matriks *state space* sistem SMIB dengan PSS yang ditala dengan RDPSO. Apabila dibandingkan, terdapat perubahan nilai frekuensi yakni nilai imajiner dari beberapa *eigenvalue*. Dapat dilihat pada *eigenvalue* ke-1, ke-2, ke-3, dan ke-4 pada sistem SMIB tanpa PSS, sisi real *eigenvalue* mengalami peningkatan nilai

redaman pada SMIB dengan PSS, dan SMIB dengan PSS yang ditala dengan RDPSO.

Dapat disimpulkan pula bahwa kedua sistem SMIB yang dilengkapi dengan PSS, baik dioptimalkan ataupun tidak, berada pada kondisi stabil karena semua bagian real *eigenvalue* bernilai negatif. Frekuensi osilasi yang dihasilkan oleh *eigenvalue* dari SMIB dengan PSS yang telah dioptimalkan adalah kurang dari 1 Hz. Frekuensi dari *eigenvalue* dapat dihitung dengan cara melakukan operasi pembagian antara sisi imajiner *eigenvalue* dan $2 \times 22/7$. Berdasarkan referensi, frekuensi osilasi yang kurang dari 1 Hz adalah frekuensi osilasi *Interarea*. Hal ini akan tergambar jelas pada respon perubahan frekuensi mesin, dan respon perubahan sudut rotor, sinyal yang tersisa pada respon frekuensi SMIB yang tertala dengan RDPSO akan lebih halus (riak-riak disepanjang waktu cuplikan menjadi sangat berkurang). Frekuensi yang bernilai $\pm 1,974$ Hz diperkirakan berasal dari frekuensi *local area*.

4.2 Analisis Damping Ratio

Tabel 4.4 adalah hasil dari penghitungan *damping ratio* dari sistem SMIB tanpa PSS.

Tabel 4.4. *Damping Ratio* Berdasarkan *Eigenvalue* pada SMIB non PSS

| <i>Eigenvalue</i> ke | <i>Damping Ratio</i> |
|-------------------------|--------------------------|
| 1 | 0,0198 |
| 2 | 1 |
| 3 | 0,0042 |
| 4 | 0,0042 |
| 5 | 0,1411 |
| 6 | 0,0495 |
| 7 | 0,0009 |
| 8 | 0,0083 |
| 9 | 0,0083 |
| 10 | 0,0083 |
| 11 | 0,0083 |

Tabel 4.5 adalah hasil dari penghitungan *damping ratio* dari sistem SMIB dengan PSS non RDPSO.

Tabel 4.5. *Damping Ratio* SMIB dengan PSS non RDPSO

| <i>Eigenvalue</i> ke | <i>Damping Ratio</i> |
|-------------------------|--------------------------|
| 1 | 0,0198 |
| 2 | 1 |
| 3 | 0,4018 |
| 4 | 0,0251 |
| 5 | 0,0251 |
| 6 | 0,0275 |
| 7 | 0,0275 |
| 8 | 0,0328 |
| 9 | 0,0009 |
| 10 | 0,0083 |
| 11 | 0,0083 |
| 12 | 0,0083 |
| 13 | 0,0083 |

Tabel 4.6 adalah hasil dari penghitungan *damping ratio* dengan menggunakan nilai penalaan optimal dari algoritma RDPSO:

Tabel 4.6. *Damping Ratio* SMIB Dengan PSS Tertala RDPSO

| <i>Eigenvalue</i> Ke | <i>Damping Ratio</i> |
|-------------------------|--------------------------|
| 1 | 0,0184 |
| 2 | 1 |
| 3 | 0,9119 |
| 4 | 0,1641 |
| 5 | 0,1641 |
| 6 | 0,1048 |
| 7 | 0,1048 |
| 8 | 0,1204 |

Tabel 4.6. *Damping Ratio* SMIB
Dengan PSS Tertala RDPSO

| | |
|----|--------|
| 9 | 0,0019 |
| 10 | 0,0008 |
| 11 | 0,0077 |
| 12 | 0,0077 |
| 13 | 0,0077 |

Dari hasil simulasi, baik SMIB dengan PSS yang telah dioptimalkan den RDPSO, maupun yang belum dioptimalkan, terdapat *damping ratio* yang berasal dari *eigenvalue* tertentu bernilai = 1. Hal ini menandakan bahwa *eigenvalue* tersebut pada permodelan ini mengalami *critically damped*, dengan nilai imajiner dari *eigenvalue* tersebut = 0. Pada saat itu, *eigenvalue* tersebut berada diambang batas sebelum terjadinya *overdamped*. Kejadian *overdamped* pada suatu *eigenvalue* akan menyebabkan resultan pada respon kestabilan, baik frekuensi, maupun sudut rotor mengalami perlambatan *settling time*.

4.3 Analisis CDI

Tabel 4.7 menunjukkan perbandingan nilai *Comprehensive Damping Index* dari sistem SMIB tanpa PSS, dengan PSS yang belum ditala dengan RDPSO, dan dengan PSS yang ditala menggunakan RDPSO.

Tabel 4.7. Perbandingan Nilai CDI

| Status | PSS non RDPSO | PSS dengan RDPSO |
|-----------|---------------|------------------|
| Nilai CDI | 3,7578 | 1,2506 |

Hasil diatas menunjukkan bahwa Nilai CDI pada sistem SMIB yang telah ditala menggunakan algoritma RDPSO mengalami perbaikan yang cukup besar dibandingkan dengan nilai SMIB dengan PSS yang belum dioptimalkan.

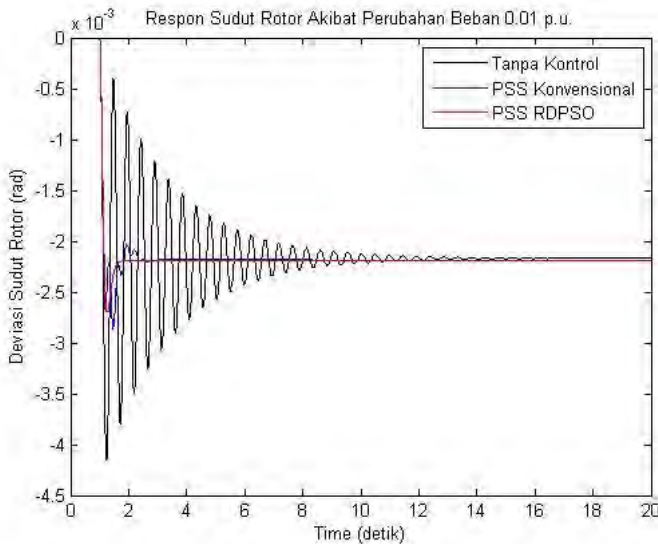
4.4 Analisis *Settling Time* dan *Overshoot*

Simulasi sistem SMIB tanpa PSS, sistem SMIB dengan PSS dan sistem SMIB dengan PSS yang ditala menggunakan RDPSO diberi gangguan sebesar 0,01 p.u dan 0,05 p.u. Respon sistem diamati dalam

waktu 20 detik. Hasil respon berupa deviasi frekuensi dan deviasi sudut rotor dijelaskan dalam sub bab berikut.

4.4.1 Deviasi Sudut Rotor Akibat Perubahan Beban 0,01 p.u, dan 0,05 p.u.

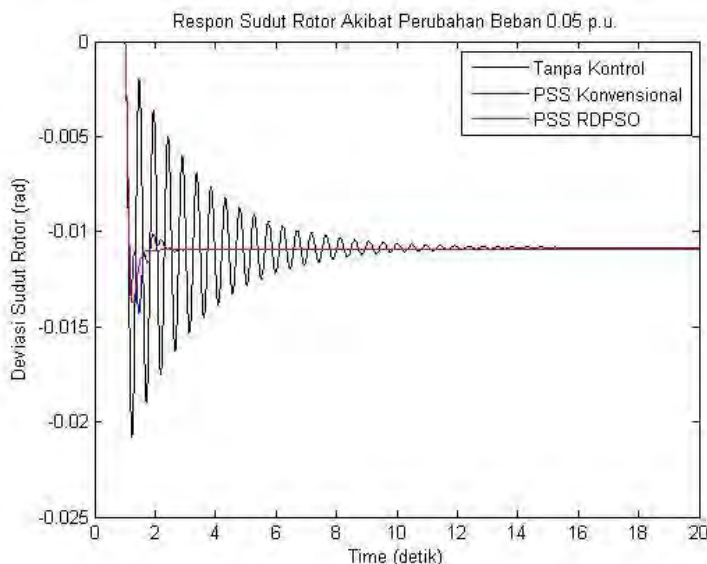
Gambar 4.2 merupakan grafik deviasi sudut rotor dari sistem akibat perubahan beban 0,01 p.u. Hasil gambar tersebut menunjukkan nilai perbandingan *overshoot* dan *settling time* dari masing-masing sistem. Sistem SMIB tanpa PSS memiliki *overshoot* dan *settling time* masing-masing sebesar -0,0042 rad dan 7,925 detik. Sistem SMIB dengan PSS memiliki *overshoot* dan *settling time* masing-masing sebesar -0,0029 rad dan 1,998 detik. Sistem SMIB dengan PSS yang ditala dengan PSO memiliki *overshoot* dan *settling time* sebesar -0,0027 rad dan 1,506 detik.



Gambar 4.2. Deviasi Sudut Rotor Pada Perubahan Beban 0,01 p.u.

Respon perubahan sudut rotor pada saat gangguan 0,05 p.u menunjukkan nilai perbandingan *overshoot* dan *settling time* dari masing-masing sistem. Sistem SMIB tanpa PSS memiliki *overshoot* dan *settling time* masing-masing sebesar -0,0208 rad dan 7,925 detik. Sistem SMIB dengan PSS memiliki *overshoot* dan *settling time* masing-masing sebesar

-0,0143 rad dan 1.998 detik. Sistem SMIB dengan PSS yang ditala dengan RDPSO memiliki *overshoot* dan *settling time* masing-masing sebesar -0,0138 rad dan 1,52 detik. Gambar 4.3 merupakan grafik deviasi sudut rotor dari sistem akibat perubahan beban 0,05 p.u.



Gambar 4.3. Deviasi Sudut Rotor Pada Perubahan Beban 0,05 p.u.

Dari kedua hasil yang didapat (perubahan beban 0,01 p.u dan 0,05 p.u), dibandingkan dengan sistem SMIB tanpa PSS, dan dengan PSS yang belum dioptimalkan, *overshoot* sistem SMIB dengan PSS yang telah ditala dengan RDPSO mengalami penurunan nilai, begitu pula dengan nilai *settling time*. Hal ini dijelaskan pada referensi [3] bahwa lama nya *settling time* dan besarnya *overshoot* dipengaruhi oleh *eigenvalue* yang terbentuk oleh sistem itu sendiri (analisis ini telah dijelaskan pada bagian analisis *eigenvalue*). Namun, pada semua gambar grafik deviasi frekuensi, dan respon perubahan sudut rotor, terlihat jelas bahwa sebagian besar sinyal osilasi yang berasal dari *local area (intermachine)* telah hilang (telah dijelaskan pada sub-bab analisis *eigenvalue*).

Dari kedua gambar respon gangguan 0,01 p.u, dan 0,05 p.u, didapatkan bahwa deviasi sudut rotor tidak kembali ke titik nol. Hal ini disebabkan oleh persamaan ayunan rotor yang dijelaskan pada bab II.

Sudut rotor berkurang dikarenakan mesin berusaha kembali ke operation poinnya dengan cara menurunkan nilai P_m kembali menjadi P_e .

P_m adalah daya mekanis pada rotor, dan P_e adalah daya yang telah dikonversikan menjadi daya listrik. Pada Tugas Akhir ini, *eigenvalue* terbentuk dari permodelan mesin *Heffron Phillips* yang sederhana, dan dari referensi yang diberikan oleh Pusat Pengaturan Beban Jawa-Bali. Nilai deviasi sudut rotor dari sistem secara lengkap dapat dilihat pada tabel 4.8.

Tabel 4.8. *Settling Time* dan *Overshoot* pada Deviasi Sudut Rotor

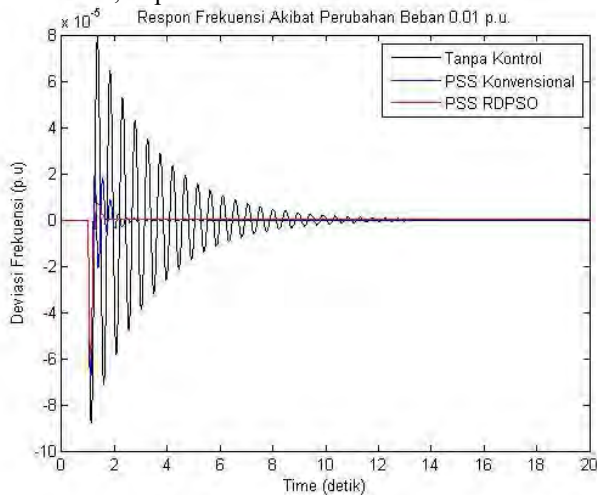
| Kasus Sudut Rotor | Perubahan 0,01 p.u | | Perubahan 0,05 p.u | |
|-------------------|-----------------------------|---------------------------|-----------------------------|---------------------------|
| | <i>Settling Time</i> (s) | <i>Overshoot</i> (rad) | <i>Settling Time</i> (s) | <i>Overshoot</i> (rad) |
| Non-PSS | 7,925 | -0,0042 | 7,925 | -0,0208 |
| Non-RDPSO | 1,998 | -0,0029 | 1,998 | -0,0143 |
| RDPSO | 1,506 | -0,0027 | 1,52 | -0,0138 |

4.4.2 Deviasi Frekuensi Akibat Perubahan Beban 0,01 p.u, dan 0,05 p.u.

Gambar 4.4 merupakan grafik deviasi frekuensi dari sistem akibat perubahan beban 0,01 p.u. Dari hasil gambar tersebut menunjukkan nilai perbandingan *overshoot* dan *settling time* dari masing-masing sistem. Sistem SMIB tanpa PSS memiliki *overshoot* dan *settling time* masing-masing sebesar 0,000087017 p.u dan 7,3778 detik. Sistem SMIB dengan PSS memiliki *overshoot* dan *settling time* masing-masing sebesar 0,000079419 p.u dan 6,593 detik. Sistem SMIB dengan PSS yang ditala dengan PSO memiliki *overshoot* dan *settling time* sebesar 0,000053017 p.u dan 7,662 detik.

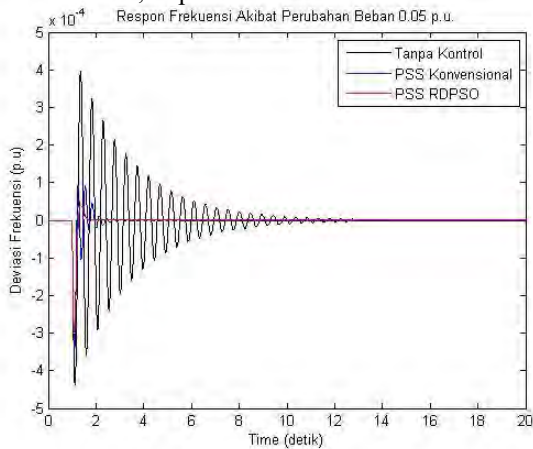
Hasil respon perubahan frekuensi akibat perubahan beban 0,05 p.u menunjukkan nilai perbandingan *overshoot* dan *settling time* dari masing-masing sistem. Sistem SMIB tanpa PSS memiliki *overshoot* dan *settling time* masing-masing sebesar -0,0284 p.u dan 10,367 detik. Sistem SMIB dengan PSS memiliki *overshoot* dan *setting time* masing-masing sebesar 0,00039709 p.u dan 10,422 detik. Sistem SMIB dengan PSS yang ditala dengan PSO memiliki *overshoot* dan *settling time* masing-masing sebesar 0,00036889 p.u dan 12,996 detik.

Gambar 4.4 adalah gambar hasil respon perubahan frekuensi akibat perubahan beban 0,01 p.u



Gambar 4.4. Deviasi Frekuensi Akibat Perubahan Beban 0,01 p.u.

Gambar 4.5 adalah gambar hasil respon perubahan frekuensi akibat perubahan beban 0,05 p.u



Gambar 4.5. Deviasi Frekuensi Akibat Perubahan Beban 0,05 p.u.

Dari kedua hasil yang didapat (perubahan beban 0,01 p.u dan 0,05 p.u), dibandingkan dengan sistem SMIB tanpa PSS, dan dengan PSS konvensional, *overshoot* dan *settling time* sistem SMIB dengan PSS yang telah ditala dengan RDPSO mengalami penurunan. Hal ini dijelaskan pada referensi [3] bahwa lama nya *settling time* dan besarnya *overshoot* dipengaruhi oleh *eigenvalue* yang terbentuk oleh sistem itu sendiri (analisis ini telah dijelaskan pada bagian analisis *eigenvalue*). Namun, pada semua gambar grafik deviasi frekuensi, dan deviasi sudut rotor, terlihat jelas bahwa sebagian besar sinyal osilasi yang berasal dari *local area (intermachines)* telah hilang (telah dijelaskan pada sub bab analisis *eigenvalue*). Performa redaman sangat dipengaruhi oleh jenis model sistm eksitasi. Nilai deviasi frekuensi dari sistem secara lengkap dapat dilihat pada tabel 4.9.

Tabel 4.9. *Time Settling dan Overshoot pada Deviasi Frekuensi*

| Kasus Frekuensi | Perubahan 0,01 p.u | | Perubahan 0,05 p.u | |
|-----------------|--------------------------|------------------------|--------------------------|------------------------|
| | <i>Settling Time</i> (s) | <i>Overshoot</i> (rad) | <i>Settling Time</i> (s) | <i>Overshoot</i> (rad) |
| Non-PSS | 13,034 | -8,75E-05 | 16,836 | -4,37E-04 |
| Non-RDPSO | 2,796 | -6,96E-05 | 3,533 | -3,48E-04 |
| RDPSO | 1,681 | -5,86E-05 | 1,8 | -2,97E-04 |

4.5 Kurva Konvergensi dan Parameter PSS Hasil RDPSO

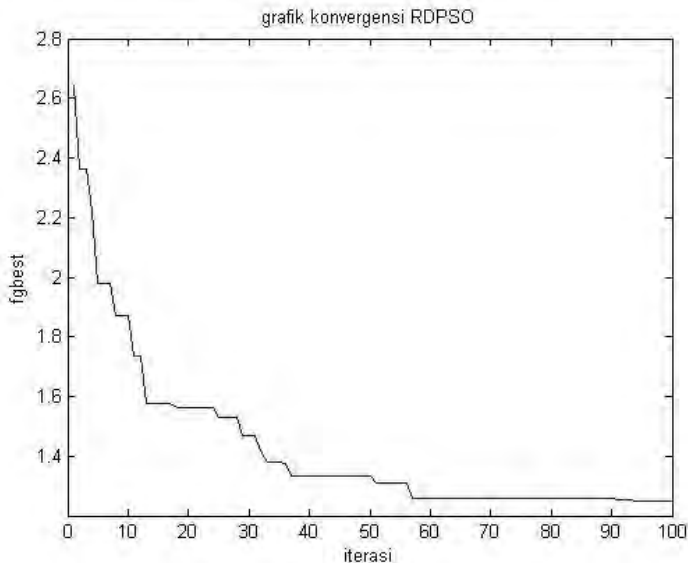
Simulasi dilakukan sebanyak 100 kali iterasi. Fungsi objektif mencapai nilai terbaik pada iterasi ke 90. Fungsi objektif terbaik menunjukkan bahwa populasi telah menemukan posisi terbaik (solusi terbaik).

Hal tersebut merupakan indikator penalaan PSS yang optimal. Parameter optimal untuk melakukan penalaan PSS pada SMIB ditunjukkan pada tabel 4.10.

Tabel 4.10. *Parameter Optimal PSS Dengan Menggunakan RDPSO*

| K_{pss} | T_w | $T1$ | $T2$ | $T3$ | $T4$ |
|-----------|--------|--------|--------|--------|--------|
| 36,1995 | 4,9346 | 0,4869 | 0,0130 | 0,0703 | 0,0122 |

Gambar 4.6 adalah gambar grafik konvergensi RDPSO pada optimisasi penalaan parameter PSS

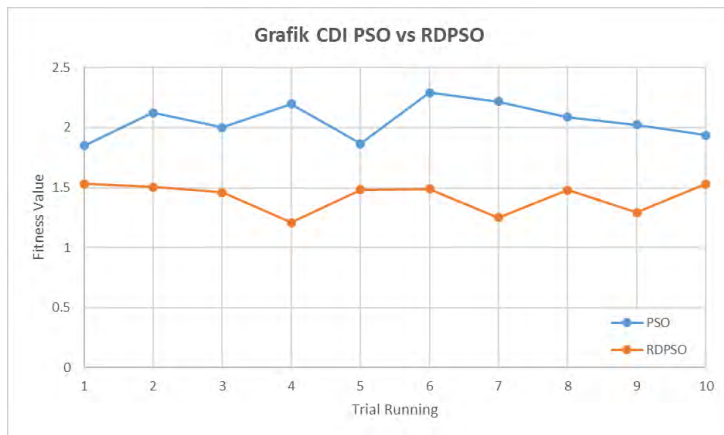


Gambar 4.6. Grafik Konvergensi RDPSO

4.6 Perbandingan Performa RDPSO dan PSO

Algoritma RDPSO adalah pengembangan dari PSO yang telah dipublikasikan pada tahun 1995. RDPSO dipublikasikan pada tahun 2015. Performa dari kedua algoritma ini dalam menyelesaikan permasalahan optimisasi khususnya penalaan parameter PSS untuk meningkatkan kualitas redaman. Perbandingan performa dari kedua jenis algoritma tersebut dibuktikan dengan cara melakukan penalaan PSS untuk SMIB Unit II Pacitan masing-masing sebanyak 10 kali. Dari hasil uji coba, RDPSO unggul dalam mencari nilai optimum dibandingkan dengan PSO. Pada kasus penalaan PSS, fungsi objektif untuk melakukan penyelesaian optimisasi adalah *Comprehensive damping Index* (CDI). CDI merupakan penjumlahan dari nilai *damping ratio* pada masing-masing *eigenvalue* sistem yang terlebih dahulu dikurangi oleh satu. Nilai optimal yang dikejar adalah nilai maksimum dari CDI. Semakin besar

CDI, semakin baik pula parameter PSS yang didapat untuk menghasilkan respon kestabilan terbaik. Gambar 4.7 adalah grafik perbandingan performa konvergensi antara PSO dan RDPSO dalam mencari nilai minimum dari CDI.



Gambar 4.7. Grafik Perbandingan Performa PSO vs RDPSO

RDPSO menghasilkan rata rata CDI yang didapat sebesar 2,059, dengan standard deviasi sebesar 0,1504, dan dengan nilai minimum CDI yang dihasilkan sebesar 1,8504. PSO menghasilkan rata rata CDI yang didapat sebesar 1,4232, dengan standard deviasi sebesar 0,1224, dan dengan nilai minimum CDI yang dihasilkan sebesar 1,2098. Tabel 4.11 adalah tabel perbandingan performa antara RDPSO dan PSO.

Tabel 4.11. Perbandingan Performa RDPSO vs PSO

| Performa | PSO | RDPSO |
|----------------------|----------|----------|
| <i>Mean</i> | 2,05973 | 1,42322 |
| <i>STD</i> | 0,150416 | 0,122471 |
| <i>Nilai minimum</i> | 1,8504 | 1,2098 |

Halaman ini sengaja dikosongkan

LAMPIRAN I

PEMROGRAMAN RDPSO

```
%% -----  
-----%%  
%PENALAAAN OPTIMAL POWER SYSTEM STABILIZER (PSS) MENGGUNAKAN  
ALGORITMA  
%PARTICLE SWARM OPTIMIZATION (PSO) DENGAN PERMODELAN SMIB  
%PADA PEMBANGKIT LISTRIK
```

```
% Supervisor I : Prof. Dr. Ir. Adi Soeprijanto, MT.  
% Supervisor II : Ir. Ni Ketut Aryani, MT.
```

```
% -----  
-----%  
% Name : Okto Fenno  
% NRP : 2212100135  
% Power System Simulation Laboratory),  
% Institut teknologi Sepuluh Nopember, Indonesia  
% Cellphone: (+62)812 3151 9369  
% Email: okto12@mhs.ee.its.ac.id & oktofenno12@gmail.com  
%% -----%%
```

```
clear all;  
close all;  
clc;  
format long;  
warning('off','all');  
tic;
```

```
j = sqrt(-1);  
Sbase = 1000;  
Pgen = 150.037/Sbase;  
Qgen = -17.403/Sbase;  
Sdaya = Pgen + Qgen*j;  
phi = angle(Sdaya);  
Vt = 1;  
R = 0.028369801368622;  
X = 0.129260477618450;  
Z = R + j*X;
```

```
Ia = Pgen/(Vt*cos(phi)); % Sudut I terhadap Vt=-(phi)  
I = Ia*(cos(-phi) + j*sin(-phi));  
Ir = Ia*cos(phi);  
Ix = -Ia*sin(phi);
```

```
Xq = 0.083699519;
```

```

Xq1 = 0.014348489;
Xd = 0.088482348;
Xd1 = 0.012937554;
r = 0.000176965; %rstator

delta = atan((Xq*Ir+r*Ix)/(Vt+r*Ir-Xq*Ix));
% Sudut antara Vt dengan sumbu Q
Id = -Ia*sin(delta+phi);
Iq = Ia*cos(delta+phi);
Vd = -Vt*sin(delta);
Vq = Vt*cos(delta);

%persamaan
E = Vq+r*Iq-Xd*Id;
Vinf = Vt-Z*I;
Vo = abs(Vinf);
alfa = angle(Vinf); % Sudut Vo
terhadap Vt
gamma = abs(alfa)+abs(delta); % Sudut Vo
terhadap sumbu Q
Eq1 = E+(Xd-Xd1)*Id;
Eqao = Eq1-(Xq-Xd1)*Id;

% Persamaan K1-K6 dari mas suharto
KL = 1/(R.^2+(Xq+X)*(Xd1+X));

K11 = Eqao*(R*sin(gamma)+(Xd1+X)*cos(gamma));
K12 = Iq*(Xq-Xd1)*((Xq+X)*sin(gamma)-R*cos(gamma));
K1 = KL*Vo*(K11+K12);
K2 = KL*(R*Eqao+Iq*(R.^2+(Xq+X).^2));
K3 = 1/(1+KL*(Xd-Xd1)*(Xq+X));
K4 = Vo*KL*(Xd-Xd1)*((Xq+X)*sin(gamma)-R*cos(gamma));
K51 = (KL*Vo*Xd1*Vq/Vt)*(R*cos(gamma)-(Xq+X)*sin(gamma));
K52 = (KL*Vo*Xq*Vd/Vt)*((Xd1+X)*cos(gamma)+R*sin(gamma));
K5 = K51+K52;
K6 = (Vq/Vt)*(1-KL*Xd1*(Xq+X))-(Vd/Vt)*KL*Xq*R;

%ubah sesuai data yang didapatkan dari P3B
%=====Other Parameter SMIB=====
%=====Parameter Mesin Sinkron=====
H = 4.114;
w0 = 2*pi*50;
D = 7;
Td0 = 8.6;
M = 2*H;
Vref = 0;

%=====Parameter ES ST4B=====

```

```

Tr = 0.01;
Kpr = 33;
Kir = 3;
Ta = 0.05;
Kpm = 1;
Kim = 0;
Kga = 0;
Vuel = 0;
Voel = 999;

```

```

%=====Parameter Gov G1=====

```

```

Kg = 10;
Tg1 = 0.5;
Tg2 = 0.1;
Tg3 = 1;
Tg4 = 1.2;
Tg5 = 1.2;
Tg6 = 1.2;
Tg7 = 1.2;
Kg1 = 0.1;
Kg3 = 0.1;
Kg5 = 0.3;
Kg7 = 0.5;
Kg2 = 0;
Kg4 = 0;
Kg6 = 0;
Kg8 = 0;
Pref = 0;

```

```

%=====Parameter PSS1A=====

```

```

Trs = 0;
A1 = 0;
A2 = 0;

```

```

% %berdasarkan abido

```

```

Constraint = [50      5      1.5      0.1  1.5      0.1;
%maksimum
              0.001  4.5      0.06      0.01  0.06      0.01];
%minimum

```

```

%batas daerah penyelesaian

```

```

sigma0      = -1;
damping0     = 0.2;

```

```

%Parameters PSO

```

```

Dim = max(size(Constraint));
Npop = 100; %aslanya 20
iter_maks = 100;

```

```
epsilon = 0.00001;
```

```
amin = 0.3;  
amaks = 0.9;  
b = 1.45;  
c1 = 2.05;  
c2 = c1;
```

```
%inisialisasi individu  
Posisi = zeros(Npop,Dim);  
for i = 1:Npop  
    for j = 1:Dim  
        Posisi(i,j) = Constraint(2,j) +  
rand*(Constraint(1,j)-Constraint(2,j));  
    end  
end
```

```
%inisialisasi kecepatan tiap partikel awal  
velocity = zeros(Npop,Dim);
```

```
%mencari eigenvalue dari posisi partikel  
for k = 1:Npop  
    %memasukkan nilai tiap individu ke PSS  
    Kpss = Posisi(k,1);  
    Tw = Posisi(k,2);  
    T1 = Posisi(k,3);  
    T2 = Posisi(k,4);  
    T3 = Posisi(k,5);  
    T4 = Posisi(k,6);
```

```
    % Mencari fitness berupa minimum CDI (Comprehensive  
Damping Index)  
    [mA, mB, mC, mD]=linmod('SMIB_pacitan_edit');  
    akar =eig(mA);  
    Jumeigen=length(akar);  
    CDI=0;  
    for nh=1:Jumeigen  
        if akar (nh)~=0  
            damping(nh)=-real(akar(nh))/abs(akar(nh));  
        else  
            damping(nh)=0;  
        end  
        CDI=CDI+(1-damping(nh));  
    end  
    J3(k,1) = CDI;  
end
```

```

Pbest = Posisi;
J3_Pbest = J3;
[gbest,line] = min(J3);
Pgbest = Posisi(line,:);
record_gbest (1,1) = gbest;
record_Pgbest (1,:) = Posisi(line,:);

%iterasi PSO
Error = 10;
iter = 2
while iter <= iter_maks %&& Error > epsilon

    %menghitung mean best (mbest)
    mbest = mean(Pbest);

    for i = 1:Npop
        for j = 1:Dim
            r = rand;
            R = rand;
            c(i,j) = c1*r/(c1*r + c2*R);
            p(i,j) = c(i,j)*Pbest(i,j)+(1-
c(i,j))*Pgbest(1,j);
            velocity(i,j) = ((amin+(iter/iter_maks)*(amaks-
amin))*abs(Posisi(i,j)-mbest(1,j))*randn)+(b*(p(i,j)-
Posisi(i,j)));
            Posisi(i,j) = Posisi(i,j) + velocity(i,j);
        end
    end

    for i=1:Npop
        for j = 1:Dim
            if Posisi(i,j) < Constraint(2,j)
                Posisi(i,j) = Constraint(2,j) +
rand*(Constraint(1,j)-Constraint(2,j));
            end
            if Posisi(i,j) > Constraint(1,j)
                Posisi(i,j) = Constraint(2,j) +
rand*(Constraint(1,j)-Constraint(2,j));
            end
        end
    end

    %mencari eigenvalue dari posisi partikel
    for k = 1:Npop
        %memasukkan nilai tiap individu ke PSS
        Kpss = Posisi(k,1);
        Tw = Posisi(k,2);
        T1 = Posisi(k,3);
        T2 = Posisi(k,4);
        T3 = Posisi(k,5);
    end
end

```

```

T4      = Posisi(k,6);

% Mencari fitness berupa minimum CDI (Comprehensive
Damping Index)
[mA, mB, mC, mD]=linmod('SMIB_pacitan_edit');
akar =eig(mA);
Jumeigen=length(akar);
CDI=0;
for nh=1:Jumeigen
    if akar (nh)~=0
        damping(nh)=-real(akar(nh))/abs(akar(nh));
    else
        damping(nh)=0;
    end
    CDI=CDI+(1-damping(nh));
end
J3(k,1) = CDI;

end

for k = 1:Npop
    if J3(k,1) < J3_Pbest(k,1)
        Pbest(k,:) = Posisi(k,:);
        J3_Pbest(k,:) = J3(k,1);
    end
end

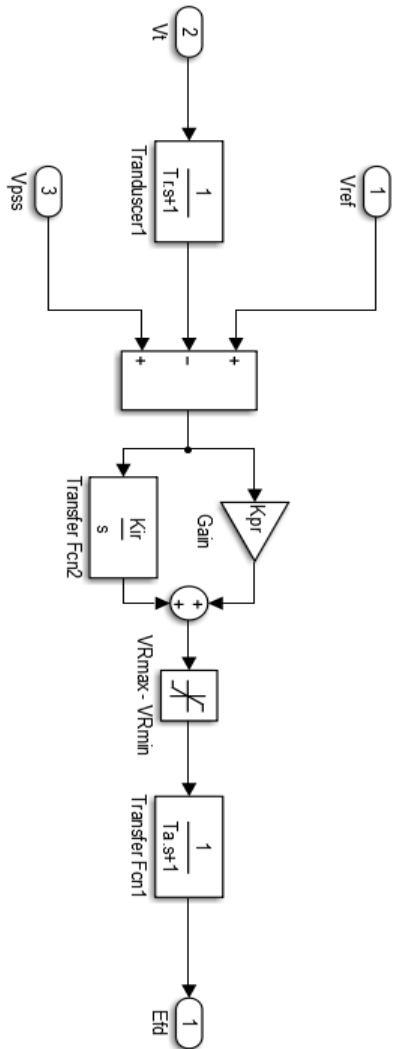
[gbest,line] = min(J3_Pbest);
record_Pgbest (iter,:) = Pbest(line,:);
record_gbest (iter,1) = gbest;
Pgbest = Pbest(line,:);
iter = iter + 1

end
%evaluasi terakhir
[val, baris] = min(J3_Pbest)
Kpss      = Pbest(baris,1);
Tw        = Pbest(baris,2);
T1        = Pbest(baris,3);
T2        = Pbest(baris,4);
T3        = Pbest(baris,5);
T4        = Pbest(baris,6);

plot(record_gbest);
title('grafik konvergensi RDPSO')
xlabel('iterasi')
ylabel('fgbest')
run_time = toc

```


LAMPIRAN II
EXCITER PSS/E ESST4B



LAMPIRAN IV

PENGHITUNGAN SETTLING TIME DAN OVERSHOOT

```
% mencari nilai overshoot dan settling time-----
-----
sampling = length(kecepatan) ;
% x = 2 tanpa pss
% x = 3 pss konvensional
% x = 4 pss pso
x=2; %biru
y=3; %kuning
z=4 ;%hijau

oke = sampling;
okea = sampling;
okeb = sampling;
while abs(kecepatan(oke, x))<= (0.05*10^-5)
    oke = oke-1;
    if abs(kecepatan(oke, x))>(0.05*10^-5)
        break
    end
while abs(kecepatan(okea, y))<= (0.05*10^-5)
    okea = okea-1;
    if abs(kecepatan(okea, y))>(0.05*10^-5)
        break
    end
while abs(kecepatan(okeb, z))<= (0.05*10^-5)
    okeb = okeb-1;
    if abs(kecepatan(okeb, z))>(0.05*10^-5)
        break
    end
end
plot(kecepatan(:,1),abs(kecepatan(:,x))); %biru
hold on ;
plot(kecepatan(:,1),0.05*10^-5*(kecepatan(:,1))./kecepatan(:,1)), 'LineStyle', '--', 'Color', 'red'); %batas toleransi

plot(kecepatan(:,1),abs(kecepatan(:,y)), 'Color', 'yellow');
plot(kecepatan(:,1),abs(kecepatan(:,z)), 'Color', 'green');
```

```

plot(kecepatan(oke+1,1),(abs(kecepatan(oke+1,x))), 'LineStyle
','o');
plot(kecepatan(okea+1,1),(abs(kecepatan(okea+1,y))), 'LineStyle
le','o','Color','yellow');
plot(kecepatan(okeb+1,1),(abs(kecepatan(okeb+1,z))), 'LineStyle
le','o','Color','green');

time_settlinga = kecepatan(oke+1,1)
time_settlingb = kecepatan(okea+1,1)
time_settlingc = kecepatan(okeb+1,1)
%-----
[valor, tos] = max(abs(kecepatan(:,x)))
waktu_overshoot = kecepatan(tos,1);

valosa = max(abs(kecepatan(:,y)));
valosb = max(abs(kecepatan(:,z)));
valor
valosa
valosb
% -----

sampling2 = length(sudut_rotor) ;
oke2 = sampling2;
oke2a = sampling2;
oke2b = sampling2;

x=2
y=3
z=4
while abs(sudut_rotor(oke2, x))<=
(abs(1.05*(sudut_rotor(sampling2,x)))) &&
abs(sudut_rotor(oke2, x))>=
(abs(0.95*(sudut_rotor(sampling2,x))))
    oke2 = oke2-1 ;
    if abs(sudut_rotor(oke2, x))>
(abs(1.05*(sudut_rotor(sampling2,x)))) &&
abs(sudut_rotor(oke2, x))<
(abs(0.95*(sudut_rotor(sampling2,x))))
        break
    end
end
while abs(sudut_rotor(oke2a, y))<=
(abs(1.05*(sudut_rotor(sampling2,y)))) &&
abs(sudut_rotor(oke2a, y))>=
(abs(0.95*(sudut_rotor(sampling2,y))))
    oke2a = oke2a-1 ;
    if abs(sudut_rotor(oke2a, y))>
(abs(1.05*(sudut_rotor(sampling2,y)))) &&

```

```

abs(sudut_rotor(oke2a, y))<
(abs(0.95*(sudut_rotor(sampling2,y))))
        break
    end
end
while abs(sudut_rotor(oke2b, z))<=
(abs(1.05*(sudut_rotor(sampling2,z)))) &&
abs(sudut_rotor(oke2b, z))>=
(abs(0.95*(sudut_rotor(sampling2,z))))
    oke2b = oke2b-1 ;
    if abs(sudut_rotor(oke2b, z))>
(abs(1.05*(sudut_rotor(sampling2,z)))) &&
abs(sudut_rotor(oke2b, z))<
(abs(0.95*(sudut_rotor(sampling2,z))))
        break
    end
end

plot(sudut_rotor(:,1),abs(sudut_rotor(:,x)));
hold on ;
plot(sudut_rotor(:,1),abs((1.05*(sudut_rotor(sampling2,x))))
*(sudut_rotor(:,1)./sudut_rotor(:,1)), 'LineStyle', '--
', 'Color', 'blue');
plot(sudut_rotor(:,1),abs((0.95*(sudut_rotor(sampling2,x))))
*(sudut_rotor(:,1)./sudut_rotor(:,1)), 'LineStyle', '--
', 'Color', 'blue');
plot(sudut_rotor(:,1),abs((1.05*(sudut_rotor(sampling2,y))))
*(sudut_rotor(:,1)./sudut_rotor(:,1)), 'LineStyle', '--
', 'Color', 'yellow');
plot(sudut_rotor(:,1),abs((0.95*(sudut_rotor(sampling2,y))))
*(sudut_rotor(:,1)./sudut_rotor(:,1)), 'LineStyle', '--
', 'Color', 'yellow');
plot(sudut_rotor(:,1),abs((1.05*(sudut_rotor(sampling2,z))))
*(sudut_rotor(:,1)./sudut_rotor(:,1)), 'LineStyle', '--
', 'Color', 'green');
plot(sudut_rotor(:,1),abs((0.95*(sudut_rotor(sampling2,z))))
*(sudut_rotor(:,1)./sudut_rotor(:,1)), 'LineStyle', '--
', 'Color', 'green');

plot(sudut_rotor(:,1),abs(sudut_rotor(:,y)), 'Color', 'yellow'
);
plot(sudut_rotor(:,1),abs(sudut_rotor(:,z)), 'Color', 'green')
;

plot(sudut_rotor(oke2+1,1),(abs(sudut_rotor(oke2+1,x))), 'Lin
eStyle', 'o')
plot(sudut_rotor(oke2a+1,1),(abs(sudut_rotor(oke2a+1,y))), 'L
ineStyle', 'o', 'Color', 'yellow')

```

```

plot(sudut_rotor(oke2b+1,1),(abs(sudut_rotor(oke2b+1,z))), 'LineStyle','o','Color','green')

time_settling2 = sudut_rotor(oke2+1,1)
time_settling2a = sudut_rotor(oke2a+1,1)
time_settling2b = sudut_rotor(oke2b+1,1)
%-----
[valos2, tos2] = max(abs(sudut_rotor(:,x)))
waktu_overshoot2 = sudut_rotor(tos2,1);
valos2
valos2a= max(abs(sudut_rotor(:,y)))
valos2b= max(abs(sudut_rotor(:,z)))

```

LAMPIRAN V

TENTANG PLTU PACITAN

PLTU Pacitan merupakan salah satu dari realisasi program percepatan pembangunan pembangkit listrik 10 MW di Indonesia. Diresmikan pada tanggal 16 Oktober 2013, PLTU Pacitan memproduksi daya berkekuatan 2 x 315 MW. PLTU Pacitan dibangun di Jalan Pacitan – Trenggalek Km 55, Desa Sukorejo, Kecamatan Sudimoro. Proyek ini bertujuan memenuhi kebutuhan listrik di Jawa Timur yang terus meningkat. Pembangunan pembangkit ini akan membantu pengembangan wilayah lintas selatan sehingga banyak investor yg akan masuk ke Jawa Timur dan akan mengurangi kemiskinan di Jawa Timur. Bahan bakar yang digunakan untuk PLTU Pacitan adalah batubara dengan kalori rendah (*Low Rank Coal*) yang didatangkan dari Kalimantan dan Sumatera. Peralatan utama untuk pembangkitan listrik PLTU Pacitan terdiri dari boiler, turbin, dan generator. Alat transportasi sumber batubara menggunakan tongkang (*barge*) dengan kapasitas 12.000 DWT (*Dead Weight Ton*) dan kapal (*vessel*) dengan kapasitas 20.000 DWT. Pembangunan PLTU ini dikerjakan oleh konsorsium Dongfang Electric dan PT Dalle Energy. Sedangkan operasi dan perawatan dikerjakan oleh PT PJBS. Untuk evakuasi dayanya disalurkan melalui GI PLTU Pacitan 150 KV menuju GI Pacitan dan/atau GI Nguntoronadi. Kebutuhan batu bara dipasok oleh PT Bukit Asam.

Beroperasinya PLTU pacitan membawa dampak positif bagi sistem kelistrikan Jawa-Bali. Beberapa aspek yang perlu diperhatikan setelah masuknya PLTU Pacitan ini adalah aspek kualitas listrik dan ekonomi. Dilihat dari aspek ekonomi, PLTU berbahan bakar batubara ini dapat menekan penggunaan PLTG yang berbahan bakar minyak yang harga listrik nya dapat mencapai 10 kali lipat harga listrik PLTU. Tentu saja, biaya produksi listrik PLN dapat semakin ditekan. Sedangkan dari aspek kualitas listrik, terutama tegangan, beroperasinya PLTU ini dapat memperbaiki profil tegangan APB Jawa Timur dan Jawa Tengah pada saat beban puncak. Data menunjukkan, sebelum PLTU ini beroperasi, jumlah GI yang mengalami ekskursi tegangan di APB Jawa Timur yang dipasok oleh *Island* Kediri sebesar 42.3 % atau 11 dari 26 gardu induk, baik sisi tegangan 150 kV, maupun 70 kV. Akan tetapi, setelah PLTU Pacitan ini beroperasi, jumlah GI yang mengalami ekskursi tegangan turun, di APB Jawa Timur menjadi 7.6% atau 2 Gardu Induk di sisi 70

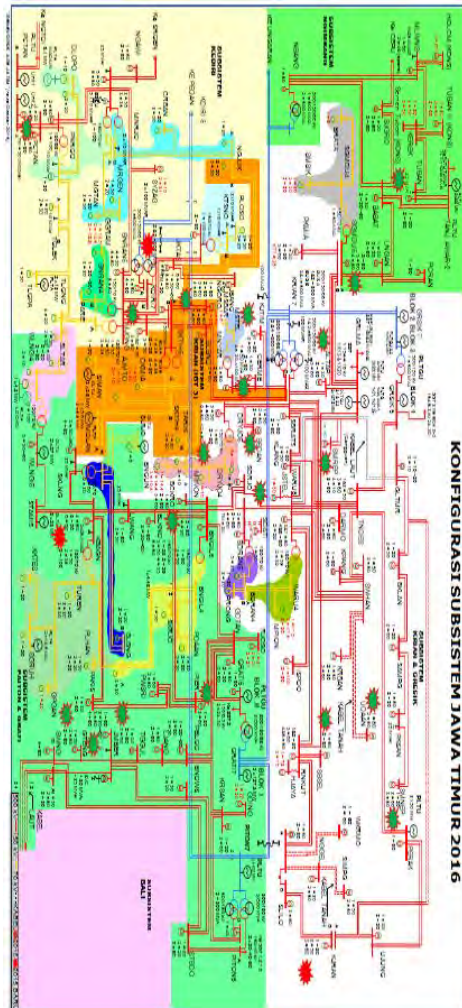
kV. Perbaikan tegangan juga cukup signifikan di APB Jawa Tengah, jika pada Januari 2013 sebelum PLTU Pacitan beroperasi ada 17 dari 90 GI 150 KV yang mengalami ekskursi tegangan, setelah PLTU Pacitan beroperasi ekskursi tegangan bias berkurang hingga menuju nilai nol.

Sistem kelistrikan 150 kV Jawa-Timur merupakan sistem yang menanggung beban yang cukup besar. Terjadinya suatu perubahan rasio permintaan beban dan total daya yang dihasilkan pada sistem ini dapat menyebabkan terjadinya osilasi pada sistem. Hal ini sejalan dengan hipotesis dari tim peneliti Pusat Pengaturan Jawa-Bali pada pasca gangguan di PLTU Pacitan pada tahun 2014. Pemadaman listrik yang terjadi di sebagian wilayah Jakarta dan Tangerang berawal dari matinya PLTU Pacitan 2 x 315 MW di Jawa Timur. Kejadian ini menyebabkan turunnya daya listrik hingga 600 megawatt (ME) di pembangkit Muara Karang dan Interbus Transformer (IBT) Kembangan II dan Penghantar Petukangan-Gandul. Osilasi yang terjadi pada pembangkit Pacitan diduga juga disebabkan oleh nilai redaman yang jelek dan pembangkit menyuplai beban yang jaraknya cukup jauh dari lokasi pembangkit.

LAMPIRAN VI

SINGLE LINE DIAGRAM SISTEM KELISTRIKAN

JAWA TIMUR



BAB V PENUTUP

5.1 Kesimpulan

Berikut kesimpulan dari tugas akhir ini :

1. Optimisasi penalaan PSS menggunakan RDPSO pada kasus Pacitan menghasilkan nilai $K_{pss} = 36,1995$, $T_w = 4,9346$, $T_1 = 0,4869$, $T_2 = 0,0130$, $T_3 = 0,0703$, dan $T_4 = 0,0122$.
2. Metode RDPSO dapat digunakan untuk menentukan koordinasi parameter PSS. Penalaan PSS dengan RDPSO pada sistem SMIB dapat memperkecil *overshoot* dan *settling time*, lebih baik dari pada sistem tanpa PSS dan sistem yang dikontrol PSS namun tidak dioptimalkan dengan RDPSO.
3. Besarnya konstanta redaman dan frekuensi gangguan dipengaruhi oleh nilai kompleks dari *eigenvalue*. Nilai *Eigenvalue* dipengaruhi oleh pemilihan model eksitasi, AVR, governor, turbin, dan *exciter*.
4. Pada tugas akhir ini, PSS berhasil menghilangkan sebagian besar osilasi. hal ini dapat ditunjukkan dengan nilai CDI sebesar 1,26, lebih baik dibandingkan dengan nilai CDI sistem yang belum diptimalkan ($CDI = 3,75$) Namun, terdapat frekuensi osilasi *interarea* (dibawah 1 Hz) tidak dapat dihilangkan dan tetap tertinggal didalam sistem.
5. RDPSO menghasilkan rata-rata CDI yang didapat sebesar 2,059, dengan standard deviasi sebesar 0,1504, dan dengan nilai minimum CDI sebesar 1,8504. PSO menghasilkan rata rata CDI yang didapat sebesar 1,4232, dengan standard deviasi sebesar 0,1224, dan dengan nilai minimum CDI yang dihasilkan sebesar 1,2098. Secara keseluruhan, Performa RDPSO lebih baik daripada PSO.

5.2 Saran

Berikut saran dari penulis untuk pembaca :

1. Penalaan parameter PSS ini juga dapat dilakukan dengan metode optimisasi yang lain.
2. Pengujian dapat dilakukan dalam kondisi gangguan lain yaitu *transient*.
3. Mahasiswa yang berminat mengembangkan topik Tugas Akhir ini dapat memperkaya konten pencarian *Req* dan *Xeq* dengan cara melakukan pencarian dengan metode *generator contribution*.

DAFTAR PUSTAKA

- [1] Adi Soeprijanto, “*Desain Kontroler untuk Kestabilan Dinamik Sistem Tenaga Listrik*”. Surabaya: ITS press, 2012.
- [2] Imam Robandi, “*Desain Sistem Tenaga Modern*”. Yogyakarta: ANDI press, 2006.
- [3] Prabha Kundur, “*Power System Stability and Control*”. New York: McGraw-Hill Inc, vol. 12, 1994.
- [4] P.M. Anderson, A. A. Fouad, “*Power System Control And stability Second Edition*”, USA: IEEE Press, 2003.
- [5] M. A. Abido, “*Optimal Design of Power–System Stabilizers Using Particle Swarm Optimization*”: IEEE Transactions On Energy Conversion, Vol. 17. No. 3, 2002.
- [6] Katsuhiko Ogata, “*Modern Control Engineering Fifth Edition*”: New Jersey: Pearson Education, Inc, 2010.
- [7] Suharto, “*Penalaan power system stabilizer (PSS) untuk perbaikan stabilitas dinamik pada sistem tenaga listrik menggunakan Bat Algorithm (BA)*”. Surabaya: Tugas Akhir Jurusan Teknik Elektro, Institut Teknologi Sepuluh Nopember, 2015.
- [8] L.J. Cai, “*Simultaneous Coordinated Tuning of PSS and FACTS Controller for Damping Power System Oscillations in Multi Machine Systems*”, IEEE Bologna PowerTech Conference, Bologna, 2003.
- [9] Pal. B, B. Chaudhuri, “*Robust Control in Power Systems*”, USA: Springer, 2005.
- [10] Rusilawati, “*Determination of Generator Steady State Stability Limit Using Losses Concept and RBFNN*”, Malaysia: UTM Press, 2015.
- [11] Jun Sun, “*Random Drift Particle Swarm Optimization Algorithm : Convergence Analysis and Parameter Selection*”, USA: Springer, 2015

RIWAYAT HIDUP



Okto Fenno dilahirkan di kota Pangkalpinang, Propinsi Kepulauan Bangka Belitung, 21 Oktober 1991, anak pertama dari pasangan Andi Chandra dan Elvaningsih Pardede. Penulis memulai jenjang pendidikannya di SD Depati Amir, Pangkalpinang. Setelah itu, penulis melanjutkan jenjang studinya di SMPN 2 Pangkalpinang hingga tahun 2006. Penulis kemudian memutuskan untuk merantau ke tanah Jawa sejak tahun 2007 dan melanjutkan studinya di Pondok Modern Darussalam Gontor Ponorogo hingga tahun 2010. Penulis sempat mengajar selama satu tahun di Pondok Modern Darussalam Gontor V Banyuwangi hingga tahun 2011. Ketertarikannya kepada sains, dan cintanya terhadap bangsa yang begitu besar, mengantarkannya untuk melanjutkan studi di Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember. Penulis sempat aktif di program beasiswa Rumah Kepemimpinan selama dua tahun, kegiatan kemahasiswaan sebagai Kepala Departemen Pengembangan Sumber daya Mahasiswa selama satu tahun, ketua pelaksana National Leadership Camp 2016, Depok, dan terdaftar sebagai asisten Power System Simulation Laboratory sebagai *coordinator of laboratory event*. Penulis dapat dihubungi melalui email oktofенno12@gmail.com